НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ
УКРАЇНИ«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ
імені ІГОРЯ СІКОРСЬКОГО»
Факультет електроніки
(повна назва інституту/факультету)

Кафедра звукотехніки та реєстрації інформації
(повна назва кафедри)

«На правах рукопису»
УДК 778.534.9

«До захисту допущено»
Завідувач кафедри
_____ Г.Г.Власюк
(підпис) (ініціали, прізвище)
“ 10 ” грудня 2018 р.

Магістерська дисертація

зі спеціальності 171 Електроніка
(код і назва спеціальності)

на тему: «Дослідження безмаркерних методів захоплення руху»

Виконала: студентка VI курсу, групи ДВ-72мп
(шифр групи)

Виноградча Ельвіра Вадимівна _____
(прізвище, ім’я, по батькові) (підпис)

Науковий керівник асистент, к.т.н., Романюк М. І. _____
(посада, науковий ступінь, вчене звання, прізвище та ініціали) (підпис)

Рецензент _____ _____
(посада, науковий ступінь, вчене звання, науковий ступінь, прізвище та ініціали) (підпис)

Засвідчую, що у цій магістерській дисертації немає запозичень з праць інших авторів без відповідних посилань.
Студент _____
(підпис)

Київ – 2018 року

Національний технічний університет України
«Київський політехнічний інститут
імені Ігоря Сікорського»

Інститут (факультет) Факультет електроніки
(повна назва)

Кафедра Кафедра звукотехніки та реєстрації інформації
(повна назва)

Рівень вищої освіти – другий (магістерський) за освітньо-професійною програмою

Спеціальність 171 Електроніка
(код і назва)

ЗАТВЕРДЖУЮ
Завідувач кафедри
_____ Г.Г.Власюк
(підпис) (ініціали, прізвище)
«10» вересня 2017 р.

ЗАВДАННЯ
на магістерську дисертацію студенту

Виноградчій Ельвірі Вадимівні
(прізвище, ім'я, по батькові)

1. Тема дисертації «Дослідження безмаркерних методів захоплення руху»,
науковий керівник дисертації Романюк Маргарита Ігорівна, к.т.н.
(прізвище, ім'я, по батькові, науковий ступінь, вчене звання)
затверджені наказом по університету від «07» листопада 2018 р. № 4114-с

2. Строк подання студентом дисертації 10.12.2018 р.

3. Об'єкт дослідження: системи безмаркерного захоплення руху

4. Предмет дослідження (Вихідні дані – для магістерської дисертації за освітньо-професійною програмою): методи та технології безмаркернеого захоплення руху, 4 вебкамери Phillips (SPC900NC), підключених до одного ПК, центральний процесор: P4 3.2 ГГц, графічний процесор, що реалізує SfS: NVIDIA Quadro 3450, бібліотека IEEE 1394, бібліотека OpenCV [DHF+], середовище розробки програмного забезпечення: Visual Studio.

5. Перелік завдань, які потрібно розробити: проаналізувати існуючі методи розпізнавання і відстеження окремих об’єктів образу людини та виділення і розпізнавання обличчя, проаналізувати програмні та апаратні засоби захоплення руху, запропонувати рішення, що дозволяють підвищити якісні характеристики безмаркерної технології захоплення руху, розробити та дослідити систему безмаркерного захоплення руху, що може бути використана для навчальних потреб

6. Перелік графічного (ілюстративного) матеріалу: 1) 43 рис, 24 табл., 1 презентація, 10 слайдів.

7. Орієнтовний перелік публікацій: 1) Дослідження можливостей безмаркерного захоплення руху з багатовидовим структурованим світлом // IX міжнародна науково-практична інтернет-конференція «Проблеми та перспективи сучасної науки», 2018 р.,- С.5-10 2) Дослідження технології доповненої реальності в освіті та перспективи її застосування при вивчені електроніки // IV міжнародна науково-практична інтернет-конференція «Наука та освіта в умовах трансформації суспільства», 2018 р., -С.41-45 3) Можливості та перспективи використання технології доповненої реальності у сучасній освіті// Науково-технічна конференція студентів, аспірантів та науковців «Сучасні проблеми застосування електронних та інформаційних технологій в телекомунікаціях, телебаченні та цифровому кінематографі», 2018 р., - С.10

8. Дата видачі завдання 10. 09. 2017 р.

Календарний план

№ з/п	Назва етапів виконання магістерської дисертації	Строк виконання етапів магістерської дисертації	Примітка
1	Написання першого розділу: Аналітичний огляд безмаркерної технології захоплення руху	15.12.2017	
2	Написання другого розділу: Апаратно-програмні засоби безмаркерного захоплення руху	30.05.2018	
3	Написання третього розділу: Особливості застосування різних методів безмаркерної технології захоплення руху	10.10.2018	
4	Написання четвертого розділу: Досладження безмаркерної системи тривимірного захоплення руху людини з використанням кількох видів з камер	09.11.2018	
5	Підготовка матеріалів до друку та оформлення пояснювальної записки	30.11.2018	
6	Підготовка та оформлення плакатів для доповіді	03.12.2018	

Студент _____ Е.В. Виноградча
(підпис) (ініціали, прізвище)

Науковий керівник дисертації _____ М.І. Романюк
(підпис) (ініціали, прізвище)

УДК 778.534.9

РЕФЕРАТ

Магістерська дисертація: 120 с., 43 рис., 24 табл., 45 джерел, 1 додаток.

ЗАХОПЛЕННЯ РУХУ, БЕЗМАРКЕРНА СИСТЕМА, ЗОБРАЖЕННЯ З ГРАДІЄНТОМ ГЛИБИНИ, 3D СКАНЕРИ, КОМП'ЮТЕРНЕ РОЗПІЗНАННЯ, ДЕРЕВА РІШЕНЬ, ФОРМА З СИЛУЕТУ.

Актуальність теми роботи полягає в тому, що комп'ютерне розпізнання образів, базуючись на аналізі зображення, та захоплення руху набуло широкого розповсюдження в різних галузях, таких як індустрія комп'ютерних ігор, в інтерактивних комп'ютерних системах, кінематографі, тощо.

Об'єктом дослідження є безмаркерні технології і способи реалізації захоплення руху з використанням різних алгоритмів розпізнавання образів.

Метою дослідження є визначення найкращого методу захоплення руху в плані компактності, швидкодії, ефективності та кінцевого результату для подальшого застосування в учбовій лабораторії кіновиробництва. Аналіз технічної реалізації з виявленням головних переваг і недоліків.

Для досягнення поставленої мети необхідно виконати такі завдання:

- проаналізувати безмаркерні технології захоплення руху для різних цілей;

- дослідити алгоритми розпізнавання образів;

- розглянути приклади реалізацій студій з використанням апаратно-програмних засобів;

- провести аналіз особливостей застосування різних методів безмаркерної технології захоплення руху;

- виконати порівняльний аналіз цих методів, виявити недоліки та запропонувати інший метод, максимально зручний та простий для користувачів.

SUMMARY

Master’s dissertation: 120 p., 43 fig., 24 tabl., 45 sources, 1 supplement.

MOTION CAPTURE, MARKERLESS SYSTEM, IMAGE DEPTH GRADIENT, 3D SCANNERS, COMPUTER RECOGNITION, DECISION TREE, SHAPE WITH SILHOUETTE.

Actuality of work is that computer image recognition based on image analysis and motion capture widespread in various fields such as industry of computer games, interactive computer systems, film and etc.

The object of the research is markerless technologies and methods of realization of motion capture using various algorithms of image recognition.

The aims of the work to determine the best method of motion capture in terms of compactness, performance, efficiency and results for later use in educational laboratory filmmaking. Analysis of technical implementation with the identification of the main advantages and disadvantages.

To achieve this goal it is necessary to perform the following tasks:

- analyze markerless motion capture technology for various purposes;
- to explore algorithms for image recognition;
- consider examples of studio implementations using hardware and software tools;
- to analyze the characteristics of various methods markerless motion capture technology;
- perform a comparative analysis of these methods, identify shortcomings and suggest another method that is most user-friendly and easy for users.

ЗМІСТ

ПЕРЕЛІК УМОВНИХ СКОРОЧЕНЬ

AR	–	Augmented reality (доповнена реальність);
CGI	–	Computer-generated imagery (зображення, згенеровані комп’ютером);
CPU	–	Central processing unit (центральний процесор) ;
EM	–	Expectation-Maximization (Очікування-Максимізація)
GPU	–	Graphics processing unit (графічний процесор) ;
KL	–	Kullback-Leibler (розбіжність Кулбака-Лейблера) ;
KPCA	–	Kernel Principal Component Analysis (основний компонентний аналіз ядра) ;
LLE	–	Locally Linear Embending (локальне лінійне вбудовування) ;
MAP	–	Maximum a posteriori estimate (максимальна апостеріорна оцінка) ;
PFSA	–	Probabilistic finite state automaton (імовірнісний кінцевий автомат стану) ;
SfS	–	Shape from silhouette (форма з силуету) ;
VH	–	Visual hull (візуальна оболонка) ;
VLMM	–	Variable-length Markov models (модель Маркова змінної довжини;
VR	–	Virtual reality (штучна, віртуальна реальність);

ВСТУП

Актуальність дослідження. Безмаркерне захоплення руху вже давно вивчається в системах комп'ютерного зору та використовуютьс не тільки у розважальних цілях для ігор та кіноіндустрії, а й в наукових цілях, найбільш у медицині. Безмаркерні технології не потребують спеціальних датчиків або спеціального костюма. Вони засновані на технологіях комп'ютерного зору і розпізнавання образів. Завдяки цьому стає можливим розпізнавання і відстеження руху людей в звичайному, не пристосованому спеціальним чином одязі, що розширює діапазон застосування подібних систем. Зазнає суттєвого спрощення рішення задачі створення 3D анімації - прискорюється підготовка до зйомок і з'являється можливість захоплення рухів (боротьба, падіння, стрибки) без пошкодження апаратних модулів системи.

Для автоматизації процесу аніматори звернулися до motion capture. Дослідники біокінетики, наприклад, Том Калверт (Tom Calvert) з Університету Саймона Фрейзера, відкрили нові можливості із застосуванням спеціальних костюмів. Був створений пристрій для захоплення рухів осіб і тіла «Waldo», що використовується для управління аватаром нінтендовского Маріо, що спілкується з публікою на виставках. Массачусетський технологічний інститут розробив свою «графічну маріонетку» на основі LED - одну з перших технологій оптичного відстеження рухів.

Спочатку захоплення рухів було виключно студійним процесом і актори працювали в порожній кімнаті, оточеній спеціальними камерами і лампами. У наші дні захоплення рухів (в тому числі і особи) прямо на знімальному майданчику стало нормою у виробництві художніх фільмів з оцифрованими персонажами.

У той час, як комерційні продукти реального часу з використанням маркерів вже доступні, безмаркерні системи, які працюють у реальному часі, залишаються відкритою проблемою, тому що багато методів не мають надійності або вимагають дорогих пристроїв і важких алгоритмів.

Проте, технології розвиваються у цьому напрямку, тому використання новітніх технологій та алгоритмів для захоплення руху підвищує шанси на

безперебійну обробку даних у реальному часі.

Метою дослідження є виявлення найбільш надійного, легкого та зрозумілого для користувача алгоритму для безмаркерного захоплення руху у реальному часі.

Для досягнення поставленої мети необхідно вирішити такі завдання:

- проаналізувати можливості та способи реалізації безмаркерних технологій захоплення руху для різних потреб;

- дослідити алгоритми розпізнавання образів;

- розглянути приклади реалізацій студій, що використовують різні апаратно-програмні засоби для захоплення руху;

- провести аналіз особливостей застосування різних методів безмаркерної технології захоплення руху;

- виконати порівняльний аналіз досліджуваних методів, виявити недоліки та розробити метод, максимально зручний та простий для користувачів.

Об’єкт дослідження – безмаркерна система та технологія захоплення руху з використанням різних алгоритмів розпізнавання образів.

Предмет дослідження – особливості застосування різних методів безмаркерної технології захоплення руху.

Методи дослідження – теоретичне дослідження захоплення руху людини, захоплення обличчя, та використання захоплення руху в AR та VR, аналіз готових систем для проведення безмаркерного захоплення руху, критичний аналіз існуючих технологій, що використовуються для захоплення руху в реальному часі.

Наукова новизна отриманих результатів: запропоновано технології, що забезпечують швидкодію та якісні результати, спрощений алгоритм, метод з мінімальним використанням обладнання, максимально зрозумілий для звичайних користувачів та який можна реалізувати для навчальних потреб.

Практична цінність отриманих результатів: запропонована система, яка працює з декількома камерами (три або більше) і одним комп’ютером. Система заснована на аналізі тривимірної фігури, обмеженні морфології людини і сегментації шкіри 3D-форми. Вона повністю автоматизована і працює в режимі

реального часу. Об’єднуючи різноманітну тривимірну інформацію, підхід є стійким до самооклюзії. Він оцінює п’ятнадцять основних суглобів людського тіла зі швидкістю більше 30 кадрів в секунду.

Апробація результатів роботи. Результати роботи були апробовані на таких конференціях:

1. IV міжнародна науково-практична інтернет-конференція «Наука та освіта в умовах трансформації суспільства» (2018).

2. IX міжнародна науково-практична інтернет-конференція «Проблеми та перспективи сучасної науки»(2018).

3. Науково-технічна конференція «Сучасні проблеми застосування електронних та інформаційних технологій в телекомунікаціях, телебаченні та цифровому кінематографі» (2018).

Публікації

1. Е.В. Виноградча. Можливості та перспективи використання технології доповненої реальності у сучасній освіті: Матеріали науково-технічної конференція «Сучасні проблеми застосування електронних та інформаційних технологій в телекомунікаціях, телебаченні та цифровому кінематографі»./ Виноградча Е.В. К.: НТУУ «КПІ ім. Ігоря Сікорського», 2018. – С. 10.

2. Е.В. Виноградча. Дослідження можливостей безмаркерного захоплення руху з багатовидовим структурованим світлом: Матеріали IX міжнародної науково-практичної інтернет-конференції «Проблеми та перспективи сучасної науки» / Виноградча Е.В. К.: НТУУ «КПІ ім. Ігоря Сікорського», 2018. – С. 5-10.

3. Е.В. Виноградча. Дослідження технології доповненої реальності в освіті та перспективи її застосування при вивчені електроніки: Матеріали IV міжнародної науково-практичної інтернет-конференції «Наука та освіта в умовах трансформації суспільства» / Виноградча Е.В. К.: НТУУ «КПІ ім. Ігоря Сікорського», 2018. – С. 41-45

1 АНАЛІТИЧНИЙ ОГЛЯД БЕЗМАРКЕРНОЇ ТЕХНОЛОГІЇ ЗАХОПЛЕННЯ РУХУ

1.1 Безмаркерна система

Нові технології та дослідження в області комп'ютерного зору ведуть до швидкого розвитку безмаркерного підходу до захоплення руху. Безмаркерні технології не вимагають спеціальних датчиків або спеціального костюма, засновані на технологіях комп'ютерного зору і розпізнавання образів. Спеціальні комп'ютерні алгоритми призначені для того, щоб система могла забезпечувала моніторінг та аналіз декількох потоків оптичного вводу та виявляти людські форми, розбиваючи їх на складові частини для відстеження Завдяки цьому стає можливим розпізнавання і відстеження руху людей у звичайному, не пристосованому спеціальним чином одязі, що розширює діапазон застосування подібних систем. Зазнає суттєвого спрощення рішення задачі створення 3D анімації - прискорюється підготовка до зйомок і з'являється можливість захоплення рухів (боротьба, падіння, стрибки) без пошкодження апаратних модулів системи. На сьогоднішній день є обмежене число безмаркерних систем, придатних для практичного використання, хоча інтенсивні дослідження в цій області проводяться з середини 90-х років [1].

Призначене для користувача програмне забезпечення для безмаркерного захоплення рухів дозволяє обійтися без специфічного обладнання, спеціального освітлення і належним чином організованого простору.

На даний момент можна виділити два типи безмаркерних систем за типом використовуваного сенсора - кольорова камера і сенсор-далекомір.

У безмаркерних системах на основі кольорової камери захоплення руху відбувається за допомогою звичайної оптичної камери і персонального комп'ютера.

Прикладом подібної системи є рішення компанії iPi Soft. Програмне забезпечення iPi Motion Capture в якості вхідних даних використовує

зображення, отримані з декількох камер, розташованих в просторі відповідно до обраної схеми розміщення.

Захоплення руху відбувається не в режимі реального часу, а на основі обробки отриманих результатів [2]. Таким чином, процес захоплення руху включає два етапи - зйомка і розпізнавання об’єктів на отриманому відеоряді. При цьому в системах даного типу висуваються істотні вимоги до умов зйомки:

- наявність рівномірного освітлення достатньої інтенсивності;

- в просторі, який потрапляє в поле зору камер, не повинно бути сторонніх об’єктів.

Крім того в момент початку зйомки людина (актор) для захоплення руху повинна прийняти еталонну позу, для розпізнавання ключових, опорних точок, використовуваних для відстеження. Даний тип систем націлений на створення основи для 3D анімації, а не безконтактного управління.

Другий тип систем безмаркерного захоплення руху для розпізнавання заснований на аналізі даних з сенсора-далекоміра (одного або декількох).

Подібне рішення реалізовано в програмних продуктах:

- OpenNI;

- Kinect SDK;

- IPi Soft таінші.

Використання сенсорів-далекомірів дозволяє істотно спростити ряд основних завдань машинного зору, використовуваних в безмаркерних системах захоплення руху, а саме - відсікання заднього фону та сегментація об’єктів на зображенні. Внаслідок цього дані рішення є менш ресурсоємними і дозволяють здійснювати захоплення руху в режимі реального часу. Крім того, використання далекомірів скорочує кількість використовуваних камер під час захоплення руху.

В основі даних систем - аналіз зображень, в тому числі для розпізнавання і відстеження окремих об’єктів отриманого образу людини (голова, плечі, лікті, кисті, коліна, ступні).

Незважаючи на те, що технологія motion capture придумана досить давно і програмне забезпечення для реалізації цієї технології добре налагоджено, при захопленні рухів виникає чимало проблем. По-перше, технологія все ще не універсальна. Також, теоретично можна робити захоплення будь-якого руху, але на практиці все ще стикаються з проблемою самооклюзії. Інша проблема - фізичний розмір області, на якій можна виконати зйомку. Об’єкт запису повинен бути в міру великим. Також система може хибно спрацювати на відбите світло або шуми. Всі ці питання потребують вивчення та аналізу, чому і присвячені подальші розділи.

1.2 Існуючі алгоритми розпізнавання образів

Під образом розуміється структурований опис досліджуваного об’єкта чи явища, представлене вектором ознак. Кожен елемент вектору являє числове значення однієї з ознак, що характеризують відповідний об’єкт. Загальна структура системи розпізнавання [3] і етапи в процесі її розробки показані на рисунку 1.1.

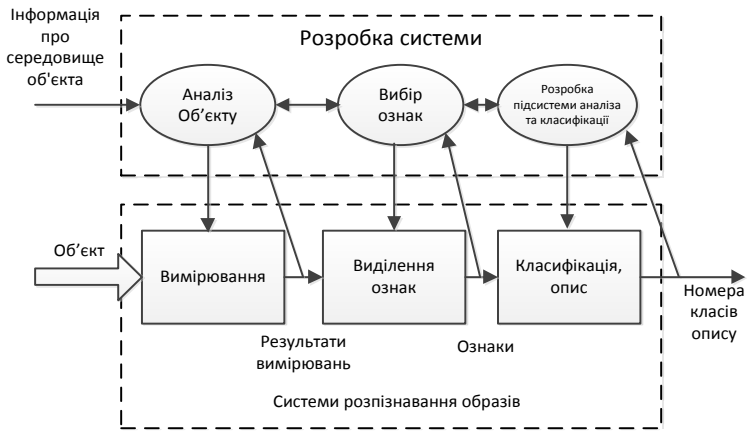


Рисунок 1.1 – Структура системи розпізнавання образів

Суть задачі розпізнавання - встановити, чи мають об’єкти дослідження фіксований кінцевий набір ознак, що дозволить віднести іх до певного класу.

Існують різні алгоритми розпізнавання образів. До основних підходів відносять:

- Розпізнання по частинах з розбиттям тіла на сегменти, що забарвлені в різні кольори.
- Позиціонування з використанням зображення просторової глибини з побудовою 3D топологічної сітки на поверхні тіла.
- Регресія, яка була основним двовимірним методом розпізнання пози людини.
- Інші методи, що використовують дерева прийняття рішень.

Розглянемо детальніше деякі з них.

1.2.1 Функції зображення з градієнтом глибини

Прості функції порівняння глибини дають лише слабкий дискримінаційний сигнал, але в поєднанні з деревами прийняття рішень вони достатньо з великою точністю усувають неоднозначності що викликаються різним виглядом різних частин тіла.

На даний піксель u, функція відгуку обчислюється як:

$$f(u|\phi) = z\left(\mathrm{u} + \frac{\delta_1}{z(\mathrm{u})}\right) - z\left(\mathrm{u} + \frac{\delta_2}{z(\mathrm{u})}\right), \qquad (1.1)$$

де параметри функції $\phi = (\delta_1, \delta_2)$ – описують 2D-зміщення пікселів δ і функції $z(u)$, що визнаає глибину в точці $u = (u, v)^T$ в конкретному зображенні. Для кожної функції виконується два пробних розрахунки для зміщення точки з глибини зображення і визначається різниця між результатами. Нормалізація зміщень на величину $1/z(u)$ гарантує, що функція відгуку є інваріантною відносно глибини: для даної точки на тілі, фіксоване просторове зміщення призведе до того, що глибинний піксель опиниться або близько, або далеко від

камери. Функції, таким чином, є просторово інваріантними відносно паралельного переносу. Якщо зміщення пікселя u' лежить на фоні, або поза межами зображення, функції $z(u')$ призначають велике додатнє постійне значення.

Під час навчання дерева рішень, зміщення δ відбираються випадково всередині коробки фіксованого розміру. Далі, якщо покласти $\delta_2 = 0$ з імовірністю 1/2. Це означає, що приблизно половина оцінених функцій є "унарними" (дивляться тільки на одну точку зміщення), а половина - "бінарними" (дивляться на дві точки зміщенння). На практиці результати здаються досить нечутливими до цього параметру.

На рис.1.2 представлені дві різні функції. Унарна функція з параметром ϕ_1 направлена вгору: формула (1.1) дає великий додатній відгук на пікселі u біля верхньої частини тіла, та значення, близьке до нуля для пікселів u у нижній частині. З аналогічних міркувань бінарна функція (ϕ_2) може розглядатися для допомоги у пошуку тонких вертикальних структур, таких як, наприклад, рука.

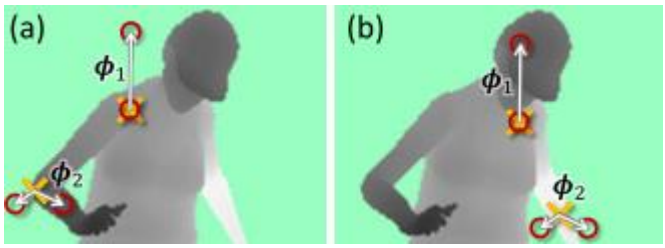


Рисунок 1.2 – Функції зображення з градієнтом глибини

Жовті хрестики на рис. 1 вказують на пікселі зображення u, які будуть класифікуватися. Червоні кола вказують на зміщені пікселі, як визначено формулою (1.1). На рис. 1.2 (a): дві зразкові функції дають великий відгук від різниці глибини (тобто значення $|f(u|\phi)|$ - велике). На рис. 1.2 (b), ті ж самі дві функції на нових ділянках тіла дають набагато менший відгук. На практиці

поєднання багатьох таких характеристик в дереві рішень дають потужний розподільчий сигнал.

1.2.2 Дерева прийняття рішень

Тіло людини здатне до величезного діапазону поз. При спільному моделюванні кількість можливих поз експоненціально залежить від кількості шарнірних з'єднань. Таким чином буває важко, або неможливо їх всіх записати.

Цю проблему можна подолати застосовуючи дерева прийняття рішень (регресійні дерева) до аналізу локальних сусідів пікселя, що будуть класифікуватися [4]. Структура дерева містить такі елементи: «листя» і «гілки». На гілках дерева записані атрибути, від яких залежить цільова функція, в «листі» записані значення цільової функції, а в інших вузлах — атрибути, за якими розрізняються події. Цільова функція — функція, що зв'язує мету (змінну, що оптимізується) з керованими змінними в задачі оптимізації. У широкому сенсі цільова функція - це математичний вираз деякого критерію якості одного об'єкту (рішення, процесу і т. д.) в порівнянні з іншим. Щоб класифікувати нову випадію, треба спуститися по дереву до листа і видати відповідне значення. Подібні дерева рішень широко використовуються в інтелектуальному аналізі даних. Мета полягає в тому, щоб створити модель, яка прогнозує значення цільової змінної на основі декількох змінних на вході. Кожен внутрішній вузол відповідає одній з вхідних змінних. Дерево може бути також «вивчено» поділом вихідних наборів змінних на підмножини, що засновані на тестуванні значень атрибутів. Це процес, який повторюється на кожній з отриманих підмножин. Рекурсія завершується тоді, коли підмножина в вузлі має ті ж значення цільової змінної, таким чином, воно не додає цінності для прогнозів.

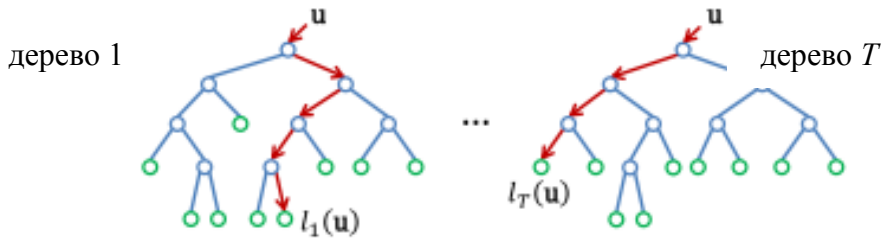


Рисунок 1.3 – Регресійний ліс з дерев прийняття рішень

Зображений на рис. 1.3 регресійний ліс складається з T дерев прийняття рішень, які видають безперервні прогнози. Кожне дерево складається з вузлів розгалуження та гілок (синій колір) і листя (зелений). На вузлах розгалуження відбуваються тести, які оцінюють особливості зображень для прийняття рішення, до якого з відгалужень, лівого або правого перейти (червоні стрілки); листя містять деякі прогнози. На вузлі розгалуження (гілці), наприклад, можна порівнювати глибину зображення, що формується пікселями, що є ближніми сусідами до контрольних. Застосування регресійних дерев дозволяє використовувати обмежений набір мокап даних та уникнути моделювання подібних, «надлишкових» поз що виникають завдяки кластеризації «далеких сусідів».

Позначення: n будь-який вузол в дереві, l – будь-який лист цього дерева. Кожен вузол розгалуження містить функцію «слабкого учня», представленого параметрами $\theta = (\varphi, \tau)$: 2D-зміщення, $\phi(\delta_1, \delta_2)$, використовується для оцінки функції зображення, описаної вище, і скалярного порогу τ. Щоб зробити прогноз для пікселя u в конкретному зображенні, починають рухатись від кореня і проходять шлях до листа, повторно оцінюючи функцію «слабкого учня»:

$$h(u;\theta_n) = [f(u;\phi_n) \geq \tau_n], \qquad (1.2)$$

де $h(u;\theta_n)$ повертає значення 0 або 1. Якщо $h(\mathrm{u};\theta_\mathrm{n})$ повертає значення 0, шлях повертає до лівого «нащадку» n, в іншому випадку він переходить до правого «нащадку». Це повторюється доки буде досягнутий лист l. Позначимо через $l(u)$ конкретний лист, який було досягнуто для пікселя u. Подібний алгоритм

застосовується для кожного пікселя для кожного дерева t, в результаті чого формується множина листів $f(u) = \{l_t(u)\}^T_{t=1}$.

1.2.3. Функції зображення з визначенням форми з силуету

Форма з силуету (SfS) - це модальність 3D-реконструкції, де доступними даними є двох динамічні проекції форми за різними видами, тобто силуетами [5].

Форма обчислюється як максимальний об’єм, сумісний з усіма силуетами. На практиці вона розраховується як перетин візуальних конусів, утворених силуетами та їх відповідними оптичними центрами, так званої візуальної оболонки (VH) [6].

Цікавість до SfS-технологій полягає в тому, що зазвичай вони забезпечують швидку і просту 3D-реконструкцію, яка не така точна, як та, що отримана стереосистемою з багатьма видами зображення, але таке наближення є достатньо точним для використання в якості основи для різноманітних комп’ютерних програм, таких як відстеження, аналіз руху людей, 3D-локалізація та навігація. Крім того, SfS особливо цікавить налаштування, коли на місці є декілька камер та / або різних об’єктів чи осіб, таких як під час спортивних подій, що створюють багато самооклюзій. У цих ситуаціях ефективність багатьох багатосторінкових стереоприкладних технологій зменшується, оскільки вони базуються на терміні фотоконтейденції, який експлуатує надмірність у різних поглядах.

Силуети, як правило, автоматично витягуються за допомогою методів вирахування фону. Проте в сегментованих силуетах часто виникають помилки, обумовлені оклюзіями у певних видах, при переміщенні фону, зміни освітлення, тіні, кольорової схожісті між переднім та фоновим зображенням, руху камери та навіть помилки калібрування. Багато з доступних методів SfS, які базуються на візуальному корпусі (VH) призведуть до неповних форм у

ситуаціях, коли силуети не узгоджуються через помилки сегментації.

Відокремлений елемент об’ємного представлення називається воксель (voxel) і зберігає інформацію про простір навколо себе. Якщо такий простір порожній, то воксель вважається порожнім, та або позначається як прозорий, або видаляється з моделі. Воксель, сусідній з порожнім і непорожнім, називається граничним. Саме граничні вокселі зберігають інформацію про поверхню моделі.

1.3 Виділення і розпізнавання обличчя

Завдання виділення особи людини в природній або штучній обстановці та подальшої ідентифікації завжди перебувала в ряду найбільш пріоритетних завдань для дослідників, що працюють в області систем машинного зору і штучного інтелекту. Тим не менше, багато досліджень, що проводяться в провідних наукових центрах усього світу протягом декількох десятиліть, так і не привело до створення реально працюючих систем комп’ютерного зору, здатних виявляти і розпізнавати людину в будь-яких умовах.

Для більшості сучасних систем автоматичного розпізнавання обличчя основним завданням є порівняння даного зображення обличчя з набором зображень облич з бази даних. Створювати можна і власний набір даних, та зазвичай використовують одну з доступних баз даних облич, наприклад: AT&T Facedatabase, Yale Facedatabase A, Extended Yale Facedatabase [8]:

Серйозною проблемою, що стоїть перед системами комп’ютерного зору, є велика мінливість візуальних образів, пов’язана зі змінами освітленості, забарвлення, масштабів, ракурсів спостереження. Колір і яскравість окремих пікселів на зображенні також залежить від великої кількості важко прогнозованих факторів. У число цих факторів входять:

1. число і розташування джерел світла;
2. колір і інтенсивність випромінювання;

3. тіні або віддзеркалення від навколишніх об’єктів.

Завдання виявлення об’єктів на зображенні ускладнюється також через більший обсяг даних, що містяться в зображенні. Зображення може містити тисячі пікселів, кожен з яких може мати важливе значення. Повне використання інформації, що міститься в зображенні, вимагає аналізу кожного пікселя на приналежність його об’єкту або фону з урахуванням можливої мінливості об’єктів. Такий аналіз може потребувати високих витрат необхідної пам’яті і продуктивності комп’ютера.

Вирішення цієї проблеми лежить в правильному виборі опису об’єктів, для виявлення і розпізнавання яких створюється система. Опис об’єкта має враховувати найбільш характерні особливості опису і бути досить повним, щоб відрізняти даний об’єкт від інших елементів навколишньої сцени. Щоб уникнути суб’єктивності при виборі потрібного опису, можна використовувати методи автоматичного вибору відповідних характеристик об’єкта.

До такого вибору відносяться:

1. вибір між 2D і 3D- представленнями сцени і об’єкта. Алгоритми, що використовують 2D-представлення, зазвичай простіші, ніж 3D-алгоритми, але в той же час вимагають великого числа різних описів, відповідних поданням об’єкта в різних умовах спостереження;

2. вибір між описом об’єкта як єдиного цілого або як системи, що складається з певної кількості взаємопов’язаних елементів;

3. вибір між системою ознак, що грунтуються на геометричних чи інших характеристиках, що описують специфіку об’єкта.

1.3.1 Захоплення руху обличчя

Захоплення руху обличчя - це процес електронного перетворення рухів обличчя людини в цифрову базу даних за допомогою камер або лазерних сканерів. Ця база даних може бути використана для створення комп’ютерної

анімації CG (комп’ютерної графіки) для фільмів, ігор або аватарів у реальному часі. Оскільки рух символів CG походить від рухів реальних людей, це призводить до більш реалістичної та детальної анімації комп’ютерного персонажа, ніж якщо анімація була створена вручну [9].

База даних захоплення руху обличчя описує координати або відносні положення опорних точок на обличчі актора. Захоплення може бути у двох вимірах (у цьому випадку процес захоплення іноді називається "відстеження виразу") або у трьох вимірах.

Двовимірне захоплення може бути досягнуто за допомогою однієї камери та меншого по вартості програмного забезпечення захоплення, наприклад Zign Track, Zign Creations. Такий підхід дає менш витончене відстеження і не здатний повністю зафіксувати тривимірні рухи, такі як обертання голови.

Тривимірне захоплення здійснюється за допомогою багатокамерних установок або лазерної системи маркерів. Такі системи, як правило, набагато дорожчі, складні та трудомісткі.

Запис руху обличчя пов’язаний із захопленням руху тіла, але є більш складним завдяки вимогам вищої роздільної здатності для виявлення та відстеження тонких виразів, можливих за невеликих рухів очей та губ. Ці рухи зазвичай менше ніж на кілька міліметрів, що вимагає ще більшої роздільної здатності і точності, а також різних методів фільтрації, ніж звичайно використовується для повного захоплення тіла.

Безмаркерна технологія використовує риси обличчя, такі як ніс, куточки губ і очей, зморшки, які слугують заміною маркерам, а потім відстежує їх. Ця технологія менш громіздка, і дозволяє актору бути більш виразним.

Технологія безмаркерного відстеження обличчя пов’язана з системою розпізнавання особи, потенційно може бути застосована послідовно до кожного кадра відео, в результаті відстеження особи. З іншого боку, деякі системи розпізнавання прямо не відслідковують вираз обличчя або навіть не відтворюють не нейтральні вирази, і тому не підходять для відстеження. І

навпаки, системи, такі як деформовані поверхневі моделі, поєднують часові відомості зі відповідним змістом і отримують більш надійні результати, і тому не можуть бути застосовані з одній фотографії.

Безмаркерне відстеження обличчя просунулося до комерційних систем, таких як Image Metrics, які застосовувались у фільмах, таких як Матриця та Загадкова історія Бенджаміна Баттона. Останній фільм використовував систему Mova, щоб захопити деформовану модель обличчя, яка потім була анімована комбінацією ручного методу та з використанням комп'ютерного зору.

Безмаркерні системи можна класифікувати за кількома відмінними критеріями:

1. відстеження 2D та 3D;
2. потреба особливої підготовки або іншої допомоги людям;
3. продуктивність у режимі реального часу (це можливо лише у випадку, якщо не потрібне навчання системи);
4. потреба у додаткових джерелах інформації, таких як проектовані шаблони або невидимі фарби, такі як, наприклад, використані в системі Mova.

На сьогоднішній день жодна система не є ідеальною у розрізі всіх зазначених вище критеріїв. Наприклад, система Neven Vision була повністю автоматизована і не вимагала ні прихованих моделей, ні тренувань на особу, але була 2D. Система Face / Off є 3D, автоматичною та реальною, але вимагає спроектованих візерунків.

1.3.2 Захоплення виразу обличчя

Використовуючи цифрові камери, вирази обличчя користувача обробляються для забезпечення головної пози, що дозволяє програмно відстежити очі, ніс та рот. Обличчя спочатку калібрується, використовуючи нейтральний вираз. Тоді, залежно від архітектури, брови, повіки, щоки і рота можна обробляти відмінності від нейтрального виразу. Це робиться, наприклад,

за допомогою пошуку по краях губ і визнаючи їх як унікальний об’єкт. Нерідко наносять макіяж для контрасту, або використовують інший спосіб зробити процес обробки швидше. Найкращі методи показують добрі результати у 90 % випадків, та такий високий відсоток вимагає великого коригування вручну, або незважання на помилки.

Персонажі, створені комп’ютером, фактично не мають м’язів, тому для досягнення однакових результатів використовуються різні методи. Деякі аніматори створюють кістки або об’єкти, які контролюються програмним забезпеченням для захоплення, і рухають їх відповідним чином, що при правильному підході дає гарне наближення. Оскільки обличчя дуже еластичні цей метод часто змішують з іншими.

Очікується, що це стане основним пристроєм введення для комп’ютерних ігор, як тільки програмне забезпечення буде доступне у вільному форматі, але апаратне і програмне забезпечення ще не існує, незважаючи на дослідження протягом останніх 15 років, яке виявилося практично придатним до вживання.

1.3.3 Алгоритми розпізнавання обличчя

В залежності від того для чого має бути використане захоплене зображення обличчя використовують різні алгоритми розпізнавання. Про деякі найпоширеніші пойде мова далі та потрібно зазначити, що незважаючи на різноманітність алгоритмів та підходів, типовий метод розпізнавання обличчя складається з трьох основних компонентів:

1. перетворення вихідного зображення в стандартний вигляд;
2. виділення ключових характеристик;
3. вибір механізму класифікації (моделювання): кластерна модель, метрика, нейронна мережа та ін..

Крім цього, побудова алгоритму розпізнавання спирається на апріорну інформацію про предметну область (в даному випадку – характеристики

обличчя людини) та коригується експериментальною інформацією, що з'являється по ходу розробки методу. Коротко розглянемо такі алгоритми розпізнання обличчя: Eigenfaces, Fisherfaces, Local Binary Patterns Histograms

Eigenfaces. В основі цього алгоритму лежить метод головних компонент (Principal Component Analysis – PCA), ідея якого полягає в тому, що масивні набори даних описуються корельованими змінними, та для для великої кількості інформації надається лише кілька значущих розмірів. PCA знаходить напрями з найбільшою дисперсією в даних, що називаються головними компонентами. Математично вказаний алгоритмм можно представити наступним чином:

Якщо позначити $X = \{x_1, x_2, \dots, x_n\}$ – випадковий вектор з набором спостережень $x_i \in R^d$. Далі потрібно визначити середнє значення μ та обчислити матрицю коваріації S

$$\mu = \frac{1}{n}\sum_{i=1}^{n} x_i \tag{1.3}$$

$$S = \frac{1}{n}\sum_{i=1}^{n} (x_i - \mu)(x_i - \mu)^{T_i} \tag{1.4}$$

Після цього знаходять власні значення λ_i та власні вектори ν_i із S

$$S\nu_i = \lambda_i \nu_i, i = 1,2, \dots, n \tag{1.5}$$

Зменшуючи власне значення, виначають власні вектори. До основних компонент спостережуваного вектора *x* даються:

$$y = W^T(x - \mu), \tag{1.6}$$

де $W = (\nu_1, \nu_2, \dots, \nu_k)$.

Побудова зображення на основі PCA здійснюється за допомогою:

$$x = W_y + \mu, \tag{1.7}$$

де $W = (v_1, v_2, \dots, v_k)$.

Вхідні вектори є відцентовані і приведені до єдиного масштабу зображення облич. Власні вектори, обчислені для всього набору зображень обличчя, називаються власними обличчами, через це метод головних компонент в застосуванні до зображень облич також називають методом власних облич (Eigenfaces) (рис. 1.4).

Рисунок 1.4 – Приклад зображення власних векторів (власні обличчя)

З рис. 1.4 видно, що власних векторів недостатньо для точної реконструкції зображення, проте, досліди показують, що 50 власних векторів може бути достатньо для кодування ключових рис обличчя.

Метод *Fisherfaces* (або інакше лінійний дискримінаційний аналіз Фішера) вивчає специфічну для класу матрицю перетворення, що дозволяє знаходити риси обличчя за відмінністю між ними. Продуктивність Fisherfaces сильно залежить і від вхідних даних. Та якщо використовувати цей алгоритм для розпізнання обличчя у погано освітлених сценах, то існує вилика ймовірність того, що, будуть знайдені невірні компоненти Це пов’язано з тим, що метод не фіксує освітлення.

Лінійний дискримінантний аналіз виконує зниження розмірності в класі та максимізує співвідношення між класів до розподілу всередині класів, замість максимального загального розсіювання. Ідея наступна: ті ж самі класи повинні класифікуватися щільно разом, а різні класи, як можна далі, в одній нижній мірі. Класичний алгоритм Фішера шукає проекцію W, що максимізує критерій сепарації класу:

$$W_{opt} = \arg\ max_W \frac{|W^T S_B W|}{|W^T S_W W|} \quad (1.8)$$

де S_B – матриця міжкласової дисперції, S_W – матриця внутрикласової дисперсії. Рішення цієї задачі оптимізації проводиться шляхом вирішення проблеми загальної власної цінності:

$$S_B v_i = \lambda_i S_W v_i \quad (1.9)$$

$$S_W^{-1} S_B v_i = \lambda_i v_i \quad (1.10)$$

Ранг S_W не більше (N-C), з N вибірками та C класами. У задачах розпізнавання образів число зразків N майже завжди є менше, ніж розмір вхідних даних (кількість пікселів). Далі проводиться лінійний дискримінантний аналіз для зменшення даних, оскільки ранг S_W більше не є одиничним. І після цього (1.8) записується як:

$$W_{pca} = \arg\ max_W |W^T S_T W| \quad (1.11)$$

$$W_{fld} = \arg\ max_W \frac{|W^T W_{pca}^T S_B W_{pca} W|}{|W^T W_{pca}^T S_W W_{pca} W|} \quad (1.12)$$

Матриця перетворень W, яка вводить вибірку в (C-1) -мірному просторі:

$$W = W_{fld}^T W_{pca}^T \quad (1.13)$$

Приклад реалізації алгоритму Fisherfaces з використаними даними з бази Yale Facedatabase представлено на рис.1.5:

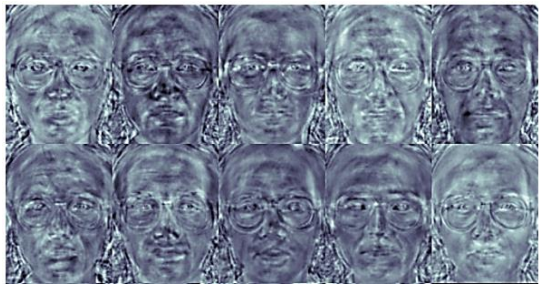


Рисунок 1.5 – Приклад реалізації алгоритму Fisherfaces

Описаний метод дає високу точність розпізнавання (близько 96 %) для широкого діапазону умов освітленості, різних виразів обличчя і наявності або відсутності окулярів. Однак залишаються нез'ясованими деякі питання, наприклад чи може метод працювати, коли в тестувальній вибірці для деяких облич є зображення тільки в одних умовах освітленості. Вищеописаний метод грунтується на припущенні про лінійну роздільність класів в просторі зображень. У загальному випадку таке припущення несправедливо. Інструментом для побудови складних подільних поверхонь пропонують нейромережеві методи.

Основна ідея алгоритму під назвою *Local Binary Patterns* полягає в узагальненні локальної структури в зображенні, порівнюючи кожен піксель з його околом. У розгляді беруть участь піксель – це центр, а також сусідні пікселів. Якщо інтенсивність центрального пікселя не менше сусіднього, то його позначають як 1 і 0, в іншому випадку. Таким чином буде отриманий двійковий номер для кожного пікселя та з 8 навколишніх пікселів вийде 2 або 8 комбінацій, які називаються Local Binary Patterns або іноді називаються кодами LBP. Перший оператор LBP, фактично використовує фіксовани 3x3 окіл.

Математично LBP можна представити як:

$$LBP(x_c, y_c) = \sum_{p=0}^{P-1} 2^p s(i_p - i_c)\,, \tag{1.14}$$

де (x_c, y_c) – центральний піксель з інтенсивністю i_c та i_p - інтенсивність сусіднього пікселя, s – функція знака, яка визначається як:

$$s(x) = \begin{cases} 1, \text{якщо } x \geq 0 \\ 0, \text{якщо } x < 0 \end{cases}, \tag{1.15}$$

Цей опис дає змогу зафіксувати дуже дрібні деталі зображення. Для заданої точки (x_c, y_c) позиція сусіда $(x_p, y_p), p \in P$ може бути розрахована за

допомогою:

$$x_p = x_c + Rcos(\frac{2\pi p}{p}) \quad (1.16)$$

$$y_p = y_c - Rsin\left(\frac{2\pi p}{p}\right) \quad (1.17)$$

де R - радіус кола, а p - кількість точок зразка.

Якщо координати точок на колі не відповідають координатам зображення, інтерполюється вхідна точка, наприклад використовується білінійна інтерполяція:

$$f(x,y) \approx [1-x\ x]\begin{bmatrix} f(0,0) & f(0,1) \\ f(1,0) & f(1,1) \end{bmatrix}\begin{bmatrix} 1-y \\ y \end{bmatrix} \quad (1.18)$$

За визначенням, оператор LBP є надійним проти монотонних перетворень сірого масштабу. Принцип дії алгоритму (див. рис.1.6) - це розділити зображення LBP на m локальних областей та зображення гістограми для кожного. Просторово-посилений вектор функції потім отримуєть шляхом об’єднання локальних гістограм. Ці гістограми називаються Local Binary Patterns Histograms.

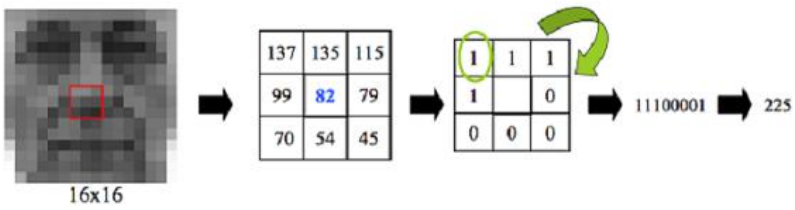


Рисунок 1.6 – Приклад дії алгоритму LBP

1.4 Безмаркерне відстеження у AR та VR

Безмаркерне відстеження – це метод позиційного відстеження –

визначення положення і орієнтації об’єкта в його оточенні. Це дуже важлива функція у віртуальній реальності (VR) та доповненій реальності (AR), що дозволяє знати область перегляду та перспективу користувача, що дозволяє віртуальному середовищу відповідно реагувати або розміщати додаткову реальность у відповідності з реальними об’єктами [10]. Для повного відстеження руху, система відстеження повинна вимірювати рух в шести ступенях свободи.

Безмаркерне відстеження використовує лише те, що датчики можуть спостерігати в навколишньому середовищі для обчислення позиції та орієнтації камери. Метод залежить від природних властивостей і може використовувати підхід на основі моделі або виконувати обробку зображень для виявлення функцій, які надають дані для визначення позиції та орієнтації.

Хоча безмаркерне відстеження - це технологія, яка, як очікується, покращить застосування VR та AR, особливо мобільних VR та AR, існуючі технологічні обмеження все ще вимагають компромісу між точністю та ефективністю. З одного боку, чим більше інформації збирає та використовує додаток, тим точніше це відстеження. З іншого боку, чим менше інформації слід врахувати обчислення, тим ефективнішим є відстеження. Ефективність - це величезна проблема для відстеження на мобільних пристроях. Доступні ресурси дуже обмежені, і відстеження не може навіть використовувати їх усіх, тому що інша частина програми потребує також потужності обробки.

1. Форма та текстура об’єкта: відстеження простіше, коли об’єкт має унікальну форму та текстуру.
2. Фон об’єкта: фоновий колір об’єкта визначає контраст між об’єктом та його середовищем. Відстеження полегшується, коли між ними виникає більший контраст.
3. Освітлення в приміщенні: інтенсивність світлового освітлення вплине на безмаркерне відстеження, оскільки камера повинна належним чином фіксувати особливості об’єктів та навколишнього середовища.

4. Світловий відбиток: оскільки освітлення є важливою функцією, яка впливає на відстеження, світлові відбиття можуть перешкоджати відстеженню.

Безмаркерне відстеження може працювати з непідготовленими середовищами. Тому цей метод є кращим для мобільних AR або VR в майбутньому. В даний час існують проблеми, які впливають на придатність та надійність системи. Вони повинні бути вирішені і тому необхідні додаткові дослідження, перш ніж цей метод відстеження стане цілком життєздатним. Вимоги до важких обчислень, непередбачувані середовища, затримка між послідовними позами все ще стримують цю технологію. Тим не менше, коли ці проблеми вирішуться, мобільні AR та VR можуть побачити більший рівень прийняття та розвитку.

Висновки до розділу

У розділі розглянуто теоретичні основи комп’ютерного розпізнання образів. Проведено огляд сучасних технологій безмаркерного захоплення руху. Наведено опис безмаркерного підходу до створення анімацій, розглянуто особливості захоплення виразу обличчя за допомогою безмаркерного захоплення руху, а також алгоритми розпізнавання образів.

Запропоновано класифікацію технологій, короткий огляд кожної з них та принципи дії. Наведені недоліки кожного з методу.

2 АПАРАТНО-ПРОГРАМНІ ЗАСОБИ ЗАХОПЛЕННЯ РУХУ

Системи для захоплення руху зазвичай включають в себе камери, апаратні модулі управління, програмне забезпечення для аналізу, обробки і виведення даних. В цьому розділі представлені найвідоміші апаратно-програмні засоби захоплення руху та наведені приклади реалізації на практиці з використанням цих засобів.

2.1 Microsoft Kinect

Kinect (оригінальна назва проекту є пристроєм введення за допомогою використання жестів для ігрової консолі Xbox 360 та ПК на базі Windows [11]. Заснований на використанні веб-камери (рис.2.1) в стилі додаткових периферійних для консолі Xbox 360, пристрій дозволяє користувачам управляти і взаємодіяти з комп'ютерним середовищем без потреби використання інших контролерів, через природний користувальницький інтерфейс за допомогою жестів і голосових команд. Проект був спрямований на розширення аудиторії користувачів Xbox 360.Kinect конкурує з контролерами WiiRemotePlus і PlayStationMove, для Wii і PlayStation 3 відповідно. Версія для Windows, була випущена 1 лютого 2012 року.

Рисунок 2.1– Зовнішній вигляд пристрою Microsoft Kinect

Microsoft випустила KinectSoftware Development Kit для Windows 16 липня червня 2011 року. Цей SDK дозволяв розробникам писати програми для

Xbox 360, а також ПК на C++ / CLI, C #, VisualBasic. NET, використовуючи функції пристрою Kinect. Одразу після випуску SDK пристрій почав набирати популярність серед незалежних розробників ігор, а також сферах 3D сканування, захоплення руху, електроніки та робототехніки.

Базується Kinect на програмній технології, розробленою компанією Rare, (дочірня компанія Microsoft GameStudios), та на технології камери для визначення відстані ізраїльського розробника Prime Sense. Ця система сканування називається структуроване світло (Light Coding) та використовує зображення для 3D реконструкції. Датчик Kinect являє собою горизонтальний брусок (див. рис. 2. 2), підключений до невеликої підставки із сервоприводом і призначена для розташування вище або нижче відео дисплея. Пристрій має RGB камеру, датчик глибини і масив мікрофонів, які забезпечують 3D-захоплення руху всього тіла, розпізнавання обличчя і можливості розпізнавання голосу. На момент запуску, розпізнавання голосу було зроблено тільки на японській та англійській мовах. Пізніше було додано французьку, німецьку, іспанську та італійську мови. Масив мікрофонів Kinect дозволяє проводити акустичну локалізацію джерела і подавлення шумів, що дозволяє пристроючути користувача навіть при наявності сторонніх звуків чи музики.

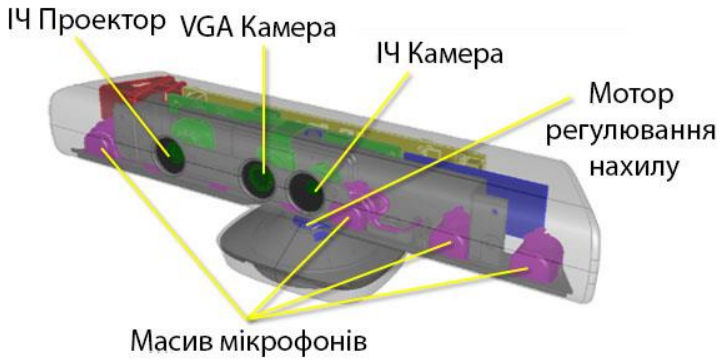


Рисунок 2.2 – Внутрішній вигляд пристрою Microsoft Kinect

Датчик глибини складається з інфрачервоного лазерного проектора в та монохромною КМОП камерою, який фіксує відеоданіу 3D за будь-яких умовах

освітлення. Діапазон чутливості датчика глибини є регульованим, і програмне забезпечення Kinectздатне виконувати автоматичне калібрування датчика спираючись на геймплеї і середовищі із гравцями, враховуючи наявність меблів та інших перешкод.

Kinect здатний одночасно виконувати відстеження до шести осіб, включаючи двох активних гравців для аналізу руху з функцією визначення 20 суглобівдля кожного гравця. Тим не менш, PrimeSense заявив, що число людей, яких пристрій може "бачити" (але не розпізнавати як гравців) обмежене лише тим, скільки поміститься в поле зору камери.

Частота кадрів різних датчиків відеозахвату Kinect складає від 9 Гц до 30 Гц залежно від роздільності відео. Типовий RGB відео потік використовує 8-бітовий VGA (640 × 480 пікселів) з колірним фільтром Баєра, але обладнання здатне робити відеозахват із роздільною здатністю 1280x1024 (при більш низькій частоті кадрів) та іншими кольоровими форматами, такі як колір UYVY. Монохромний потік глибини середовища відтворюється у роздільній здатності VGA (640 × 480 пікселів) з 11-бітною розрядністю, яка забезпечує 2048 рівня чутливості. Kinect може також записувати відео з ІЧ-камери безпосередньо (тобто до його перетворення в зображення глибини), як відео 640x480 або 1280x1024 з нижчою частотою кадрів. Датчик Kinect може визначати глибину картинки наобмеженому діапазоні 1.2-3.5 м, при використанні програмного забезпечення Xbox. Площа, необхідна для гри з використанням Kinect приблизно 6 $м^2$, хоча датчик може підтримувати відстеження через розширений діапазон приблизно 0.7-6 м, що збільшує ефективну площу використання пристрою. Датчик має поле зору у 57° по горизонталі і 43 ° по вертикалі, в той час як сервопривід дозволяє нахиляти датчик на 27° вгору або вниз. Мікрофонний масив має чотири мікрофонні капсули і працює з кожним каналом обробки 16-бітного звуку з частотою дискретизації 16 кГц.

Таблиця 2.1 – Характеристики пристрою Microsoft Kinect

Назва параметру	**Значення**
Поле зору камери	43° у вертикальній площині та 57° у горизонтальній
Вертикальний діапазон нахилу	Від -27° до +27°
Формати кадрів відео потоку	RGB камери (формат RGB/Bayer/YUV): • 1280 × 720 пікселів при 12 Гц • 640 × 480 пікселів при 30 Гц • 640 × 480 пікселів при 15 Гц ІЧ камери: • 640 × 480 пікселів при 30 Гц
Формати кадру потоку глибини	• 640 × 480 пікселів при 30 Гц • 320 × 240 пікселів при 30 Гц • 80 × 60 пікселів при 30 Гц
Аудіо формат	16 кГц, 24 біт, моно сигнал, імпульсно-кодова модуляція (PCM)
Характеристики звукових входів	Чотири мікрофонні капсули із 24-бітним аналого-цифровим перетворювачем (АЦП) і вбудована у Kinect система обробки сигналів (у тому числі погашення акустичного шуму і відлуння)
Характеристики акселерометру	2G/4G/8G акселерометр налаштований для діапазону 2G, з точністю до 1°.
Підключення	USB 2.0
Підтримувані платформи	Xbox 360/Xbox One Microsoft Windows

Оскільки мотор нахилу Kinect вимагає більше енергії живлення, ніж USB порти здатні забезпечити, пристрій використовує спеціальний роз’єм USB для об’єднання зв’язку з додатковим джерелом живлення. Перероблена модель Xbox360S включає спеціальний порт AUX для розміщення роз’єму, в той час як

більш старі моделі Xbox, а також усі персональні комп’ютери вимагають спеціальний кабель, який розділяє на окремі з’єднання USB канал і лінії живлення. Живлення самого перехідника здійснюється від мережі за допомогою адаптера змінного струму.

2.1.1. Принцип роботи системи камер

Система Kinect застосовує метод регресії безпосередньо до необробленого зображення з градієнтом глибини, без проміжних перетворень. Вона є застосовна до загальних типів рухів (не є обмеженою щодо конкретної діяльності) а також має можливість локалізувати видимі та приховані сполучення тіла.

Завданням апаратних засобів Kinect є обрахування відстані від пристрою до кожного пікселя у полі зору камери. Для обрахування зображення глибини Kinect поєднує в собі метод структурованого світла з двома класичні методами комп’ютерного зору: глибина з фокусу, і глибина зі стереозображення.

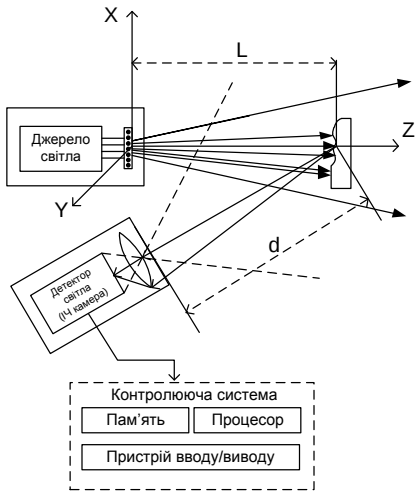


Рисунок 2.3 – Принцип роботи системи камер Microsoft Kinect

У якості проектованого шаблону Kinect використовує інфрачервоне зображення псевдовипадково розташованих цяток. Даний шаблон складається

із трьох типів точок, різних за розміром, які у сукупності утворюють повну картину проекції.

– Перша область: дозволяє отримати зображення глибини поверхні для близьких об’єктів (0,8 - 1,2 м) із високою точністю.
– Друга область для об’єктів на відстані 1,2 - 2,0 м із середньою точністю.
– Третя область призначена поверхні далеких об’єктів (2,0 - 3,5 м) із низькою точністю.

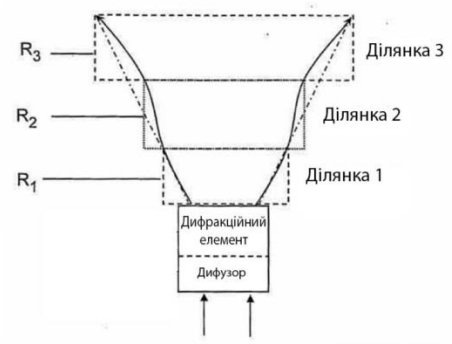


Рисунок 2.4 – Принцип роботи інфрачервоного проектору

Для проекції цяток пристрій використовує спеціальний "астигматичний" об’єктив з різною фокальною відстанню у X та Y напрямах [12]. Проміні світла пропущеного через такі лінзи приймають форму еліпса, орієнтація якого змінюється в залежності від відстані до точки відбиття.

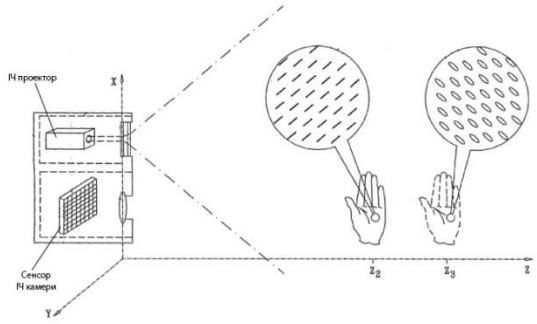


Рисунок 2.5 – Деформація цяток в залежності від відстані

Після аналізу відхилення маркерів відбитої проекції від оригінального шаблону пристрій створює зображення глибини із декількома рівнями.

Наступною дією є приведення зображення глибини до більш простого вигляду із лише одним градієнтом.

В результаті обробки зображення з градієнтом глибини можна побудувати скелет тіла, та "натягнути" на нього оболонку будь-якого героя з бібліотеки образів (вбудовані функції).

Також, користуючись інформацією наданою отриманим зображенням з градієнтом глибини можна побудувати тривимірну модель об’єкту (хмару точок) у кольоровому або монохромному варіантах

2.1.2. Засоби розробки програмного забезпечення для Microsoft Kinect

Засоби розробки програмного забезпечення, а також документація та ресурси для програмування на мовах C++ / CLI, C#, VisualBasic. NET, для пристрою Kinect надані сайтом Microsoft DeveloperNetwork.

Kinect SDK для Windows пропонує розробникам спеціалізовані бібліотеки програмного забезпечення та інструменти, які дозволяють використовувати усі методи захоплення даних пристрою Kinect. Сам пристрій і бібліотеки програмного забезпечення взаємодіють із програмою, як показано на рис. 2.6.

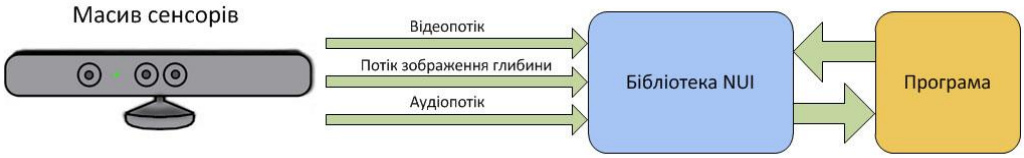


Рисунок 2.6 – Взаємодія апаратного та програмного забезпечення з програмою

Користувальницький інтерфейс (NUI) є ядром Kinect для Windows API. Через нього можливо отримати доступ до наступних даних датчика:

-Аудіо дані які були захоплені з аудіо потоку.

-Кольорове зображення і графічне представлення даних про глибину, захоплених звідеопотоку та потоку зображення глибини.

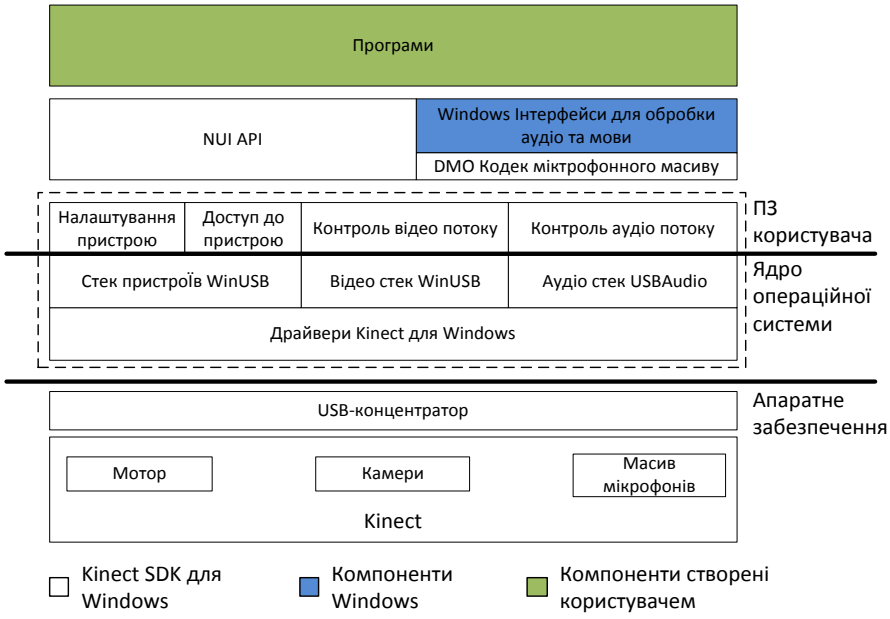


Рисунок 2.7 – Архітектура засобів розробки програмного забезпечення Microsoft Kinect

2.2 Система IPI Soft

В системі iPi Soft маркери не використовуються: це занадто довго і складно. Тільки підготовка і калібрування маркерної системи займає близько години, тобто студію захоплення руху доводиться резервувати на цілий знімальний день – а це коштує тисячі доларів. В студіях Motion Capture, де застосовуються маркерні системи, часто у всіх ролях знімаються одні й ті ж актори, і потрібен спеціальний одяг під маркери, а наборів цього спецодягу завжди обмежена кількість.

Підготовка до роботи iPi Desktop Motion Capture проходить набагато швидше [13]. До одягу також існують певні вимоги, але вони набагато простіші в реалізації. Оптимальний варіант – це чорний верх, синій низ (джинси) та чорні черевики, при цьому і верх і низ мають бути однотонними, без візерунків і без блискіток.

Перший етап – зйомка відео одночасно з різних точок. Стереокамери не підійдуть для цієї задачі, оскільких їх стереобаза порівняно невелика, через що

не вдається створити карту глибини потрібної точності.

Другий етап – відновлення тривимірної сцени. Оскільки отримується зображення кожного пікселя з різних точок, за допомогою триангуляції, є можливість відновлення тривимірного зображення. Основна складність роботи на цьому етапі – в тій частині технології, що відповідає за розпізнавання образів. Програма повинна ідентифікувати людину і її частини тіла. Тому дуже важливо встановити стільки камер, щоб актора завжди було видно з усіх боків. Абсолютно достатня кількість камер – 8. Чотири з них знаходяться в усіх кутах приміщення, ще 4 – між кожними двома сусідніми камерами.

Одна з основних проблем всіх систем Motion Capture – це необхідність збереження великих об'ємів інформації, що вони генерують на етапі зйомок, особливо, якщо стоїть задача створення "хмари точок". В першу чергу знімається велика кількість багатокамерного відеоматеріалу, а в другу –система починає прораховувати тривимірний скелет моделі.

Апаратна реалізація цього методу потребує комп'ютера з потужною відеокартою з відповідним програмним забезпеченням iPi.

2.3 Приклади використання Kinect та IPI Soft

2.3.1 Проект з двома датчиками глибини Kinect

Нижче наведено приклад реалізації захоплення руху і анімування персонажа за допомогою MoCap даних, отриманих з використанням iPi Mocap Studio і двох сенсорів (датчиків глибини, рис.2.8) [14].

Рисунок 2.8 – Датчики глибини MS Kinect 2

Для цього знадобляться:

1. 1-2 сенсора Kinect (або ASUS Xtion або PrimeSense Carmine 1.08);
2. iPi Studio, iPi Recorder;
3. Autodesk Motionbuilder.

Встановлюємо iPi Studio і iPi Recorder, підключаємо Kinect. Більш докладно про це можна дізнатися на wiki ipisoft. Для того щоб отримувати MoCap дані з 2х сенсорів, необхідно записати відео для калібрації.

Сенсори можуть бути розташовані в 2х конфігураціях:

1. Кут між сенсорами 60-90 градусів:

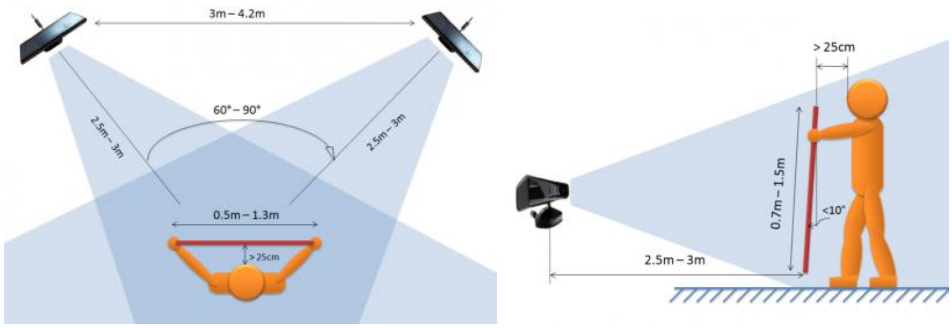


Рисунок 2.9 – Калібрація з розташуванням сенсорів із кутом 60-90^{o}

2. Кут між сенсорами 110-180 градусів:

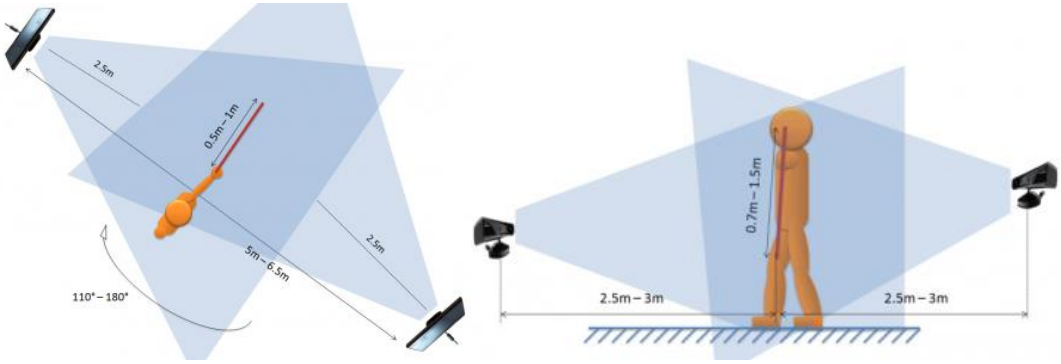


Рисунок 2.10 – Калібрація з розташуванням сенсорів із кутом 110-180^{o}

Для калібрації використовується картонка / дощечка розміром від 0,5 м. в ширину (рекомендується 1м. - 1,3 м.) і від 0,7. в довжину (рекомендується 1 -

1,5 м.) Калібраціонна поверхня повинна бути добре видна з кожного сенсора. Так неправильно:

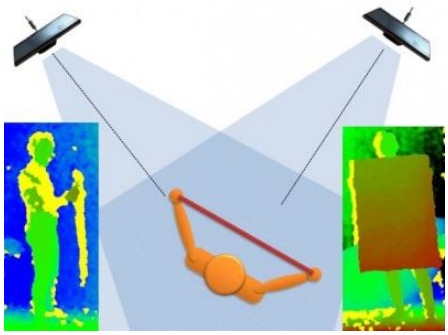


Рисунок 2.11 – Неправильне положення людини для калібрації

Жовтий колір на карті глибини означає, що сенсор не може розрізнити глибину в цих точках. Потрібно, щоб їх було якомога менше.

Важливо: якщо змінити положення сенсорів їх треба буде заново калібрувати.

1. Запускаємо iPi Recorder, вибираємо 1-2 Kinect як пристрій для запису;
2. Далі програма аналізує фон, в області видимості сенсорів повинні бути тільки нерухомі об’єкти;
3. Натискаємо на Start, встаємо і тримаємо картонку так, щоб вона була видна з кожного сенсора. Потрібно поводити картонкою в різні боки, при цьому її поверхня повинна бути видна з обох сенсорів.
4. Натискаємо Stop;
5. Відкриваємо iPi Studio -> New project -> вибираємо відео яке щойно записали -> Calibration project -> calibrate based on 3d plane -> Чекаємо -> Scene -> Save Scene.

Коли процес калібрації пройшов успішно, можна приступати до запису рухів і імпорту анімації.

1. Запускаємо iPi Recorder;

2. Далі програма аналізує фон, в області видимості сенсорів повинні бути тільки нерухомі об’єкти;
3. Натискаємо на Start, встаємо в Т-позу на 2-3 секунди, відтворюємо рух який хочемо записати;
4. Натискаємо Stop;
5. Відкриваємо iPi Studio -> New project -> вибираємо відео яке щойно записали -> Action project -> якщо відео записувалося з декількох камер потрібно буде відкрити scene отриману раніше -> Вибираємо кадр з хорошою Т-позою -> дивимося як почати MoCap;
6. Натискаємо Export -> import target character -> імпортуємо biped з 3dmax Тепер при перегляді відео поруч з чоловічком, що повторює ваші рухи, буде видно скелетик;
7. Натискаємо Export animation і зберігаємо файл у форматі fbx.

Анімація персонажа

1. Запускаємо Motionbuilder -> Window -> Actor / Character controls
2. Відкриваємо файл з fbx анімацією (File -> Open);
3. Далі нам потрібно зробити characterization, також в motionbuilder є learning movies (adding animation);
4. Window-> Asset Browser -> Templates -> Перетаскуємо значок Character на скелетик -> characterize -> biped;
5. Перетягуємо в робочий простір свого персонажа або з Asset Browser -> Merge -> No animations або File -> Merge -> в правій колонці прибираємо всі галочки;
6. У Character Controls вибираємо свого персонажа, перетаскуємо рухи скелета на рухи персонажа, натискаємо Animation -> Plot All -> Plot;
7. Зберігаємо чи можемо імпортувати персонажа в unity.

2.3.2 Проект з камерами Multiple PlayStation Eye

Для багатократної конфігурації PlayStation Eye (рис. 2.12), вам потрібно як мінімум 13 футів на 13 футів простору (4 метри на 4 метри). При меншому просторі, актор просто не поміститься в поле зору камер.

Для 640 на 480 роздільною здатністю камери, область захоплення може бути 7 на 7 метрів. Цього має бути достатньо для захоплення рухів, як біг, танці і т.д.

Рисунок 2.12 – Камери Sony PS3 Eye

Використовуючи зелений або синій фон можна поліпшити результати, але ви не зобов’язані використовувати фон, якщо у вас є прийнятний офіс або домашня обстановка зі стінами світлого кольору і яскравим освітленням. Як і в попередньому прикладі, записуємо відео за допомогою програми IPI Recorder. Вона підтримує запис з камер Sony PlayStation Eye, датчики глибини (Kinect) і DirectShow сумісні з веб-камер (USB і FireWire).

Рекомендується записувати усі відео в максимально можливій частоті кадрів. Висока частота кадрів допомагає зменшити розмитість і захопити дрібні деталі руху. Максимально можлива частота кадрів для камери Sony PlayStation Eye 60 кадрів в секунду. Sony рекламує камери PlayStation Eye, як здатна захоплювати в 120 кадрів в секунду, але більш ніж 60 частоти кадрів FPS дає занадто багато шуму в датчику камери PlayStation Eye. Двоядерний процесор повинен бути досить швидким для запису відео з 4-х камер з роздільною здатністю 320 на 240 та зі швидкістю 60 кадрів в секунду. Чотирьохядерний процесор рекомендуються для запису при роздільній здатності 640 на 480 та зі швидкістю 60 кадрів в секунду. Якщо є тільки двоядерний процесор, можливо,

буде потрібно налаштувати нижню частоту кадрів і / або низьку якість стиснення, щоб мати можливість записувати відео з роздільною здатністю 640 на 480. При використанні 6 камер з роздільною здатністю 640 на 480 зі швидкістю 60 кадрів в секунду, рекомендується чотирьохядерний процесор з тактовою частотою 2,0 ГГц (або краще). Також потребується додатковий контролер USB.

Рекомендована конфігурація для установки 3-х камер є півколом:

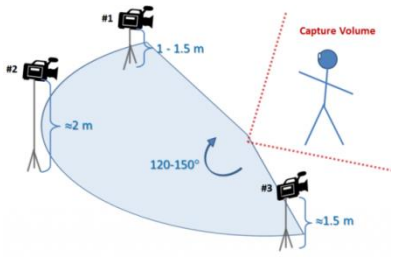


Рисунок 2.13 – Розташування 3-х камер в півколі

Можна встановити 4 камери в півколі або конфігурації повного кола, в залежності від доступного простору. Підвищити точність можна шляхом розміщення однієї з камер високо над землею (наприклад, 3 метра).

Рекомендована конфігурація для установки 4-камери в півколі:

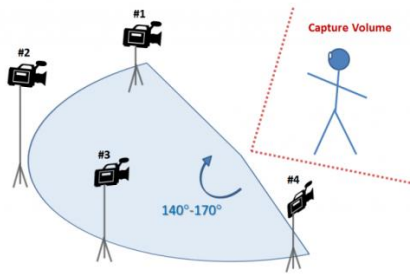


Рисунок 2.14 – Розташування 4-х камер в півколі

Камери Sony PlayStation Eye не мають стандартний гвинт штатива, так що доведеться використовувати якесь спеціальне рішення. Найпростіший підхід полягає в фіксації камери на штатив за допомогою липкої стрічки.

2.3.3 Проект з трьома датчиками глибини

Починаючи з версії 2.5.0.158, IPI Mocap Studio включає експериментальну підтримку потрійної конфігурації датчика глибини. Це допомагає вирішувати проблеми прикусу, таким чином, що дозволяє записувати більш складні рухи і відстежувати двох акторів з датчиками глибини. Якість можна порівняти з 6 камерами PS Eye, але область захоплення менше (приблизно 2,5 на 2,5 метра).

Використання потрійної конфігурації глибина датчика аналогічна конфігурації датчика Dual Depth, за винятком таких аспектів:

1. Потрійної конфігурація глибини датчика вимагає Standard Edition. В інших виданнях експорт відключений для проектів зі зміною датчика потрійної глибини.

2. Запис з трьох датчиків глибини вимагає, щонайменше, три окремих USB контролерів 2,0 / 3,0 до ПК, схожих на запис з 6 камер PS Eye.

3. Калібрування потрійної конфігурації датчика глибини може бути виконане тільки маркером, що світиться, як ліхтарик. Це означає, що, в калібрування відео, потік кольору повинен бути записаний разом з глибиною даних: тільки режим «(глибина + колір)» має бути використаний.

В ідеалі датчики глибини повинні бути розміщені в вершинах рівностороннього трикутника зі сторонами, рівними приблизно 6 метрів (рис. 2.15) На практиці це не завжди може бути досягнуто за рахунок доступних розмірів і конфігурації простору, кріплення позицій, USB і / або довжини шнура живлення, і т.д.

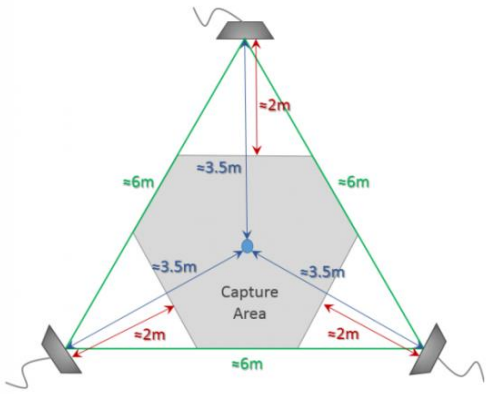


Рисунок 2.15 – Ідеальне розміщення датчиків

Таким чином, позиція датчика може трохи відрізнятися від ідеальної (рис. 2.16).

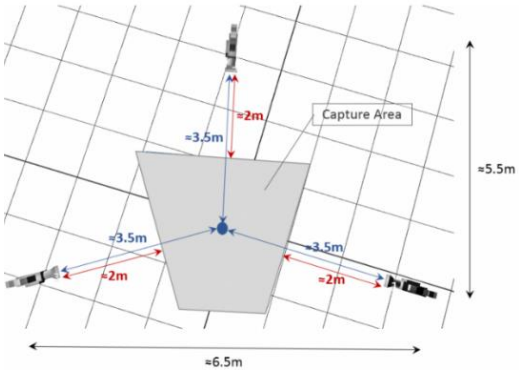


Рисунок 2.16 – Реальне розміщення датчиків

Процедура калібрування майже така ж, як з подвійною глибиною калібрування датчика за допомогою ліхтарика.

2.4 Memoji від Apple та Samsung AR Emoji

2.4.1 Memoji від Apple

Face ID - це нова функція розпізнавання обличчя, розроблена Apple [15].

Щоб зробити Face ID можливим на iPhone X, Apple довелося розміщувати багато нового обладнання в крихітну область у верхній частині дисплея. Ця

область дублюється «системою камери TrueDepth», і вона складається з верхнього підсвічування, інфрачервоної камери, передньої камери, точкового проектора, датчика наближення, датчика навколишнього освітлення, динаміка і мікрофона. Коли користувач заглядаєте в свій iPhone X, запускається система камери TrueDepth.

Щоб обробити всі дані особи, Apple розробила новий Aion Bionic neural engine. Це чіп A11 компанії Bionic з вбудованим нейронним двигуном. Animoji використовує систему камер TrueDepth, що застосовується для Face ID, а також Aion Bionic chip, щоб захоплювати і аналізувати більше 50 різних м’язових рухів на обличчі. Потім він відображає вирази обличчя в різних емодзі для створення Animoji.

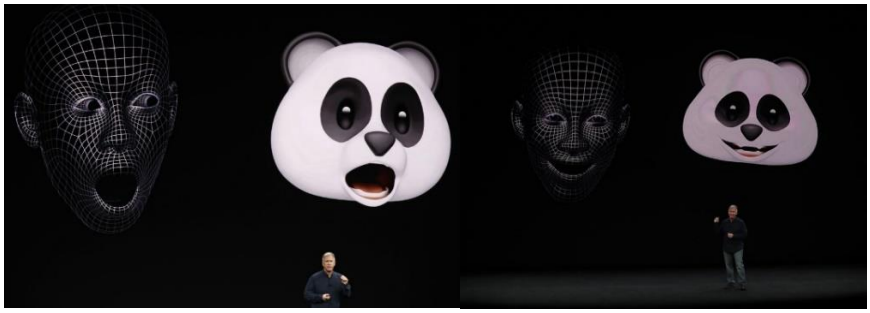


Рисунок 2.17 – Приклад відтворення мімики людини мімикою емодзі панди

Після поновлення програма може не тільки автоматично додати до тексту свиню, кота, тигра або панду замість смайлика, а й емодзі з обличчям власника пристрою. Якщо людина захоче створити свій емодзі, то зможе вибрати форму обличчя, колір шкіри, очі і їх колір, рот, ніс, вуха, аксесуари, головні убори, зачіску, міміку, поворот голови і багато іншого.

В iPhone X використовується розробка Microsoft. Kinect використовує кілька технологій для створення 3D зображення хорошої якості. Одна з них називається структуроване світло - пристрій проектує заздалегідь відомий візерунок і використовує машинне навчання створює 3D-сцену. В новому

iPhone X використовується схожая схема - проектор точок створює візерунок, а інфрачервона камера його читає.

У Kinect також встановлений модуль звичайної кольорової камери, який служить для створення карти глибин. Щоб визначити наскільки далеко розташовані об’єкти, аналізується зображення. Об’єкти, які надто далеко - зовсім розмиті.

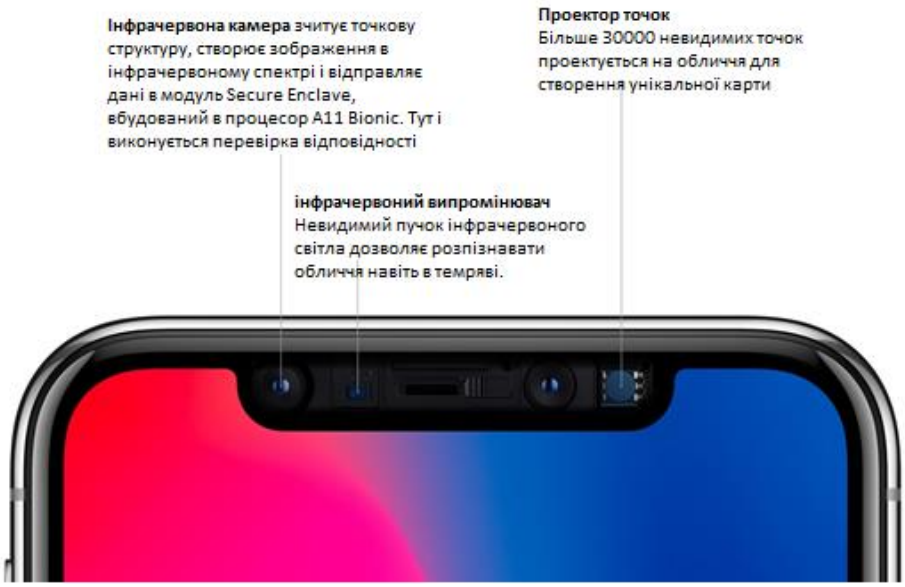


Рисунок 2.18 – Передня панель смартфону

В новому iPhone використовується ІК випромінювач, який дозволяє фронтальній камері розрізняти зображення навіть у повній темноті (інфрачервона камера використовується для читання точок проектора).

Є ще кілька технологій, що застосовуються в Kinect. У пристрої використовуються астигматичні лінзи – вони мають різну горизонтальну і вертикальну фокальну відстань. Це дозволяє обробляти один піксель двічі, що покращує якість зображення.

У будь-якому випадку обробка всіх даних зав’язується машинним навчанням та нейронними мережами. Для навчання нейронних мереж було використано тисячі прикладів положення тіла у випадку з Kinect і тисячами людей людей – у випадку з Apple iPhone X.

2.4.2 Доповнена реальність в Samsung

У новому Samsung Galaxy S9/S9+ з’явилася можливість створити анімовану копію себе, яка допоможе передати емоції.

Функція Samsung AR Emoji працює на основі фронтальної 8Мп камери з діафрагмою f/1.7. Відкрити свого персонажа дуже швидко можна через клавіатуру або в повідомленнях. Їм можна поділитися в форматі GIF або PNG, а також у вигляді стікера в підтримуваних додатках. Можно відправляти повідомлення з селфімодзі, що повторюють міміку і рухи.

Фактично Galaxy S9 робить те ж саме, що і iPhone X: за допомогою фронтальної камери сканує обличчя користувача. Однак у S9 основний акцент робиться на персоналізації вражень. Отриманий мультяшний чоловічок на екрані AR Emoji-копія обличчя користувача на всі 100%.

2.4.3 Відмінності селфімоджі (Samsung) від анімоджі (Apple)

У новому Galaxy S9 компанія Samsung скопіювала одну з головних функцій iPhone X [16]. Відмінності селфімоджі від анімоджі, наступні:

1 Точність

Функція Animoji використовує складну систему TrueDepth на iPhone X, щоб створювати об’ємну карту обличчя. Проте, що система занадто складна, але завдяки цьому вона максимально точна.

Samsung використовує тільки фронтальну 8Мп камеру Galaxy S9 з діафрагмою f/1.7. Через це персонаж AR Emoji не так точно копіює руху обличчя. Крім того, функція часто створює зовсім несхожого на користувача персонажа.

2 Можливість налаштування

На iPhone X можна вибрати тільки готового персонажа з 12 на даний момент доступних, який запише голос і скопіює вираз обличчя користувача. З

оновленням iOS 11.3 персонажів стане більше.

AR Emoji можна налаштовувати, змінюючи зачіску свого персонажа, одяг, тон шкіри і окуляри. Замість власного зображення, можна вибрати Міккі або Мінні Маус, які повторять вираз обличчя користувача, а потім їх можна буде відправити своїм друзям і близьким. З Animoji такого робити не можна, тому в даному випадку перевага у AR Emoji.

3 Можливість поділитися

Щоб створити Animoji, потрібно зайти в Повідомлення. Потім записати відео до 10 секунд і відправити його тільки тим користувачам, у яких є пристрій з iOS. Також Animoji можна поділити в форматі GIF в соціальних мережах.

Поділитися AR Emoji набагато легше. Після налаштування персонажа він зберігається у вигляді колекції з 18 GIF-файлів, які легко відкрити через клавіатуру в будь-якому додатку через яким їм можна поділитися. Крім того, ділитися AR Emoji можна в форматах GIF і PNG.

Недоліки селфімоджі

Головна проблема - необхідність у високій швидкості роботи. На створення подібних моделей в кінематографі витрачається декілька днів, тоді як в Galaxy S9 на це є пара секунд. За словами представника компанії, для створення якісного аватара співробітникам знадобиться близько 7 хвилин.

Ще один фактор - спосіб відстеження рухів лицьових м’язів. iPhone X використовує технологію 3D-сканування, тоді як в розпорядженні Galaxy S9 тільки 2D-модель. Це означає, що програмне забезпечення фіксує менше деталей, через що версія Samsung не дотягує до конкурента.

Висновки до розділу

У розділі розглянуто апаратно-програмні засоби для захоплення руху. Описано роботу пристрою Microsoft Kinect та наведено його характеристики.

Проведено огляд системи iPi Soft та наведено приклади реалізації студії з використанням дитчиків глибини Kinect, камер Multiple PlayStation Eye з відповідним програмним забезпеченням iPi.

Розглянуто нові функції розпізнавання обличчя, розроблені Apple та Samsung. Наведено їх характеристики, порівняння та недоліки.

3 ОСОБЛИВОСТІ ЗАСТОСУВАННЯ РІЗНИХ МЕТОДІВ БЕЗМАРКЕРНОЇ ТЕХНОЛОГІЇ ЗАХОПЛЕННЯ РУХУ

3.1 Спостереження людської пози в реальному часі з використанням найближчої апроксимації основного зображення компонентів ядра

В даному розділі розглянуто та досліджено безмаркерну технологію захоплення руху в режимі реального часу на основі некаліброваних синхронізованих камер. Для вивчення оптимального різноманіття поз людського руху за допомогою основного компонентного аналізу ядра (Kernel Principal Component Analysis (KPCA)) використовуються тренувальні набори реальних рухів, зафіксованих на базі маркерних систем. Після навчання, нові силуети, раніше невидимих акторів проектуються через два колектора за допомогою реконструкції локально лінійного вбудовування (Locally Linear Embedding (LLE)). Вихідна поза генерується шляхом апроксимації прообразу (зворотне відображення) реконструйованого вектора LLE з мнговиду поз.

Найважливішим чинником дорогої вартості виводу поз людини є висока розмірність вихідного простору (що становить 19 суглобів або 57 ступенів свободи в даній установці). Щоб пом’якшити вичерпний пошук у просторі многовиду поз, виведення поз може бути обмежене до нижчого просторового многовиду.

Це можливо через високий ступінь кореляції в русі людини, і може бути визначено за допомогою різноманітних методів навчання, таких як локально лінійного вкладення (LLE), Isomap [16], або технології Kernel, що засновані на напів-визначеному вставленні (SDE) [17].

Проблема з LLE, Isomap і SDE (на основі багатовимірного масштабування) полягає в тому, що вони не визначені для нових точок, які не були в зразку [18]. Для того, щоб визначити відповідне подання нового вводу в просторі вкладеного многовиду, вхід потрібно додати до навчального комплексу та всього многовиду. Це не є ідеальним для виведення руху людини

в реальному часі на основі проекції многовиду. У даному методі використовується більш ефективна методика, заснована на аналізі основних компонентів ядра (KPCA) [19], яка визначена для точок поза вибірки.

Використовується KPCA для вивчення двох представлень функціональних просторів (рис. 3.1), які виводяться з штучних силуетів і відносних скелетних спільних позицій однієї загальної сітчастої моделі людини. За допомогою LLE після тренувань нові силуети невидимих акторів (і невидимих поз) проектуються через два колектора [20]. Захоплена поза потім визначається шляхом розрахунку попереднього зображення [21] проектованих силуетів.

Невід'ємною перевагою KPCA є його здатність пригнічувати шуми вхідних зображень перед обробкою, як показано в [19]. Проте, не існує попередніх робіт з відокремлення силуетів людини для захоплення руху за допомогою проекції KPCA. Цей метод може бути застосований до безмаркерного захоплення руху та дозволяє техніці виводити порівняно точні пози з зашумлених невидимих силуетів, використовуючи лише одну штучну модель тренування людини. Обмеження цього підходу полягає в тому, що дані силуету будуть проектуватися на підпростір, орієнтований на навчання поз, тим самим обмежуючи вихід в межах цього підпростору. Це обмеження не серйозне, оскільки система може навчатися правильним підготовленим набором даних, якщо є попередні знання про тип рухів, які плануються захопити.

Метод включає в себе впровадження нової техніки для безмаркерного захоплення руху на основі регресу між двома функціональними підпросторами, визначеними через KPCA. Вводиться нова концепція зняття силуетів, яка дозволяє попередньо невидимим (тестовим) силуетам проектуватися на підпросторі, а отже, дозволяючи виводити результат з використанням єдиної тренувальної моделі (що також призводить до значного зменшення розміру тренування та часу виводу). Для відображення від силуетів до підпросторової

пози, замість використання стандартної або надійної регресії, в технології KPCA використовуються значення, визначені з LLE.

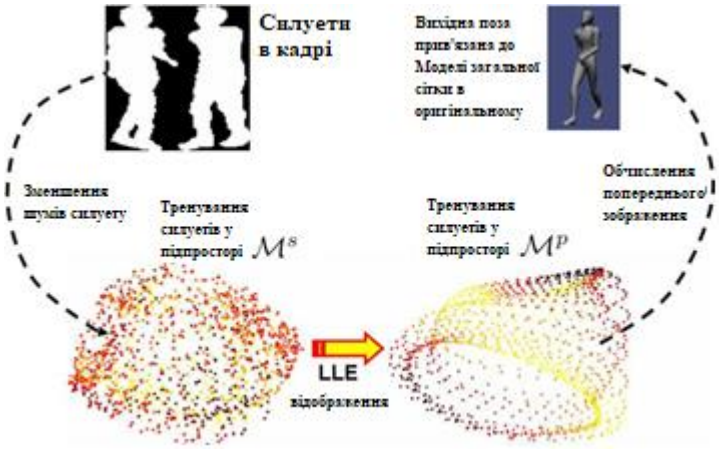


Рисунок 3.1 – Огляд технології KPCA, що базується на основі безмаркерної техніки захоплення руху

Налаштовуються параметри ядра для силуету, щоб оптимізувати відображення силуету пози, шляхом мінімізації помилки реконструкції LLE. Нарешті, шляхом відображення силуетів у многовиді поз, можна обмежити пошуковий простір до нижнього розмірного підпростору, одночасно використовуючи добре встановлений та оптимізований попередній образ (зворотне відображення) апроксимації, такий як алгоритм з фіксованою точкою. Дана методика, має такі переваги:

1. Замість того, щоб вивчати окремі многовиди для кожної точки зору, вивчається єдиний комбінований многовид з кількох точок огляду (обертається навколо вертикальної осі), це дає можливість точно визначити курсовий кут та інформацію про позу людини в один крок (і уникати явного пошуку кількох многовидів).
2. Технологія відображення, заснована на KPCA, також добре узагальнює невидимі моделі та силуети з невидимих кутів огляду.

3. Методика здатна проектувати шумові вхідні силуети (що зашумлені і лежать поза многовиду навчальних силуетів) на людину, що дохволяє використовувати єдину загальну навчальну модель та скоротити час виводу.

3.1.1 Безмаркерне захоплення руху: проблеми відображення

Пози людини кодують, використовуючи відносні спільні центри *x*,

$$x = [p_1, p_{2\ldots}p_n]^T, x\epsilon R^{3n}, \tag{3.1}$$

де p_k являє собою вектор позиції $[x, y, z]^T$ *k*-го суглоба (відносно спільного центру його джерела). Будь-яка методика, така як KPCA і регресія на основі стандартної лінійної алгебри, зрештою, не буде працювати при застосуванні до векторів, що складаються з кутів Ейлера, оскільки вони потенційно можуть задати однакове обертання 3D-суглоба в різні місця в векторному просторі. Щоб уникнути цих проблем, у даному методі показуються послідовності руху до загальної сітки моделі (рис. 3.1 - вгорі праворуч) та аналізується відносний вектор положення *x* спільної структури внутрішнього скелета над мчасовою рамкою анімації. Вивчаються многовди поз M^p (рис.3.1 - внизу праворуч) із набору тренувальних положень, закодованих у формулі 3.1. Аналогічним чином, для простору силуету попередньо обробляється синхронізовані зчеплені силуети до ієрархічного дескриптора форми π_i, використовуючи техніку, подібну до пірамідного ядра збігу [22].

З попередньо підготовленого тренувального набору вивчається силует колектора M^s (рис. 3.1 - внизу ліворуч) і налаштовується система, щоб мінімізувати помилку відновлення LLE при відображення від M^s до M^p . Під час захоплення нові силуети невидимих акторів (рис. 3.1 - зверху ліворуч) проектуються через два підпростори, перед тим, як відстежуватись до простору виведення пози, використовуючи апроксимацію попереднього зображення.

3.1.2 Вивчення многовидів поз через KPCA

Для того, щоб інтегрувати KPCA до вивчення многовидів поз M^{p}, спочатку кодується кожна позиція навчання. Кожна поза вектора x_i нелінійно відображена через додатну напіввизначену функцію ядра $k_p(x_i, x_j)$, яка визначає нелінійне співвідношення між двома векторами положення x_i і x_j. Ядро переносить вектор до функціонального простору H^{tr} так, що

$$k_p(x_i, x_j) = \left(\phi(x_i) \cdot \phi(x_j)\right), \phi: X^{tr} - H^{tr} \tag{3.2}$$

це скалярний добуток в просторі ознак. Коли в комплекті з навчальним набором X^{tr} складається з *N* зразків, KPCA кожного вектора x_1 , ..., x_N в наборі до функціонального простору відображається як $\Phi(x_1), \ldots, \Phi(x_N)$ і лінійні заготівки PCA на карті векторів. KPCA проекція нової точки *x* на *k*-ту головну вісь v_P^k в просторі функції пози можна виразити неявно через ядра, як

$$\left(v_P^k \cdot \phi(x)\right) = \sum_{i=1}^{N} \alpha_i^k \left(\phi(x_i) \cdot \phi(x)\right) = \sum_{i=1}^{N} \alpha_i^k k_p(x_i, x) \tag{3.3}$$

де α являє собою набір власних векторів централізованої матриці ядра. У даному методі використовується радіальна основа гауссівського ядра $k_p(x_i, x) = exp^{-\gamma_p\{(x_i - x)^T (x_i - x)\}}$ через наявність добре розробленого та перевіреного алгоритму наближення до зображень. Існують два вільних параметра, які потребують налаштування: γ_p евклідовий коефіцієнт масштабу відстані, і η_p оптимальне число проекцій головної осі, щоб зберегти в просторі функції пози. Позначимо проекцію KPCA (на головній осі першої η_p) x_i, як v_i^P,

де $$v_i^P = \left[\left(v_p^1 \cdot \phi(x_i)\right), \ldots, \left(v_P^{\eta_p} \cdot \Phi(x_i)\right)\right]^T, \forall v^p \in R^{\eta_P} \tag{3.4}$$

3.1.3 Налаштування параметра пози за допомогою аппроксимації попереднього зображення

Для того, щоб зрозуміти, як оптимально налаштувати параметри KPCA γ_p і η_p для многовидів поз M^p , потрібно відзначити, як M^p наведено на рис.3.1 (внизу праворуч). У контексті KPCA, якщо кодувати кожну позу як x і її відповідний проектований вектор KPCA, як v^P , зворотне відображення від v^P до x зазвичай називають відображенням попереднього зображення. Оскільки нове введення спочатку буде відображене з M^s в M^p , доведеться визначити його зворотне відображення (попереднє зображення) з M^p до початкової позиції простору. Тому налаштовуємо γ_p і η_p, використовуючи крос-валідацію, щоб мінімізувати функцію витрат реконструкції попереднього образу $C_p = \frac{1}{N}\Sigma_{i=1}^{N}\|x_i - x_i^+\|^2$, де x_i^+ є попереднім зображенням v_i^P. Цікавою перевагою використання наближення до зображень для відображення є його властива здатність подавлення шуму, якщо вхідний вектор (що обурений шумом) лежить поза чистим тренувальним різноманіттям поз.

3.1.4 Налаштування параметрів силуету за допомогою оптимізації LLE

Подібно до M^p, існують також два вільних параметра (γ_s і η_s) для налаштування підпростору силуету M^s. Використовується та сама концепція, що й у попередньому пункті, але замість цього налаштуються параметри γ_s і η_s для оптимізації відображення LLE силуету-пози. Це досягається шляхом мінімізації функції реконструкції LLE $C_s = \frac{1}{N}\sum_{i=1}^{N}\left\|v_i^p - \Sigma_{j=1}^{k} w_{ij}^s v_j^p\right\|^2$, де j показує k сусідів v_j^p. Масштаб значення w_{ij}^s в цьому випадку є ваговим коефіцієнтом v_i^s, який може бути закодований v_j^s, використовуючи евклідову відстань в M^s. Для

того, щоб переконатися, що налаштовані параметри добре узагальнюються з невидимими новими входами, система почне навчатися за допомогою перехресної перевірки на навчальному наборі. Під час захоплення, отримавши нове зображення з дескриптором π силуету, система неявно продемонструє його через настроєне ядро k_s, щоб отримати w^s (у M^s). Прогнозований вектор потім перетворюється на M^p, визначаючи ваги w^s, що мінімізує функцію помилки реконструкції LLE $\varepsilon(v^S) = \left|\left|v^s - \Sigma_{j=1}^{k} w_j^s v_j^s\right|\right|^2$. Це можна ефективно виконувати шляхом вирішення лінійної системи рівняння:

$$\sum_j G_{ij} w_j^s = 1, \text{ де } G_{ij} = (v^s - v_i^s) \cdot \left(v^s - v_j^s\right), \tag{3.5}$$

і повторно масштабувати ваги, щоб підсумувати до одного. З w^s, можливо знайти v^p, представлення пози підпростіру v^s, наступним чином: $v^p = \Sigma_j w_j^s v_j^p$, з якої можна визначити захоплену позу, знайшовши її відповідний попередній образ.

3.1.5 Кількісні експерименти з штучними даними

Щоб протестувати метод кількісно, використовують нові рухи (подібні до тренувального набору) для анімації невидимих сітчастих моделей та захоплення їх відповідних штучних силуетів для використання як контрольних зразків тесту. Використовуючи набір тренажерів зі спіральним проходженням з 343 прикладами, можна визначити нові пози при середній швидкості ~ 0,104 секунди на кадр на Pentium TM 4 з процесором 2,8 ГГц. Вихідний потік попереднього зображення порівнюється з оригінальною позою, яка була використана для створення штучних силуетів (рис.3.2 - центр). На цьому етапі модель тестової сітки відрізняється від моделі тренувальної сітки, і всі тестові

зображення виходять з невидимого кута огляду або пози. Для 1260 невидимих тестових силуетів з різних кутів повороту можливо досягти точних реконструкцій оригінальної пози із середньою похибкою 2,86° за суглоб (рис.3.2 - справа) для скелета з 57 ступенями свободи. Що стосується тестування з шумними даними, додавався шум солі та перцю до одного набору даних і результати досягли середньої похибки в 3,42° в суглобі (збільшення менш ніж 0,6 ◦ в суглобі).

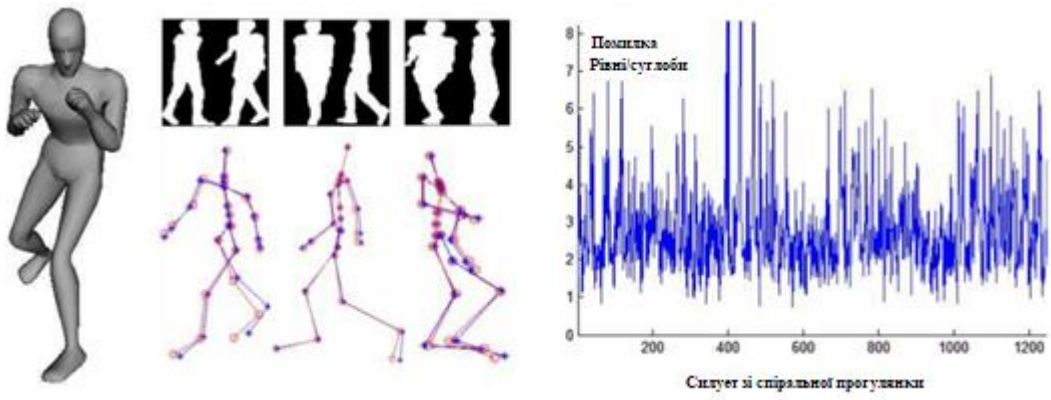


Рисунок 3.2 – (Ліворуч) Загальна модель, що використовується в навчанні. (По центру) Порівняння захопленої пози (червоні точки - суглоби) з оригінальними позами (сині зірочки "*"), які використовуються для генерації штучних тестових силуетів. (Праворуч) Помилка сюжету для руху по спіралі.

3.1.6 Якісні експерименти з реальними даними

Для тестування з реальними даними було обрано декілька спіральних рухів ходьби та застосовано просте віднімання фону. Встановлено дві перпендикулярні камери (з такими самими параметрами, як і тренувальні камери) без точних вимірів. Завдяки простоті налаштування та сегментатора, шуми від тіні та різного світла були помітними в тестових послідовностях. Вибрані результати показані на рис. 3.3. Також було додано штучно шум з солі

та перцю до реальних силуетів для перевірки надійності системи (рис. 3.4). Здатність захоплювати рух в цих різних умовах демонструє надійність техніки.

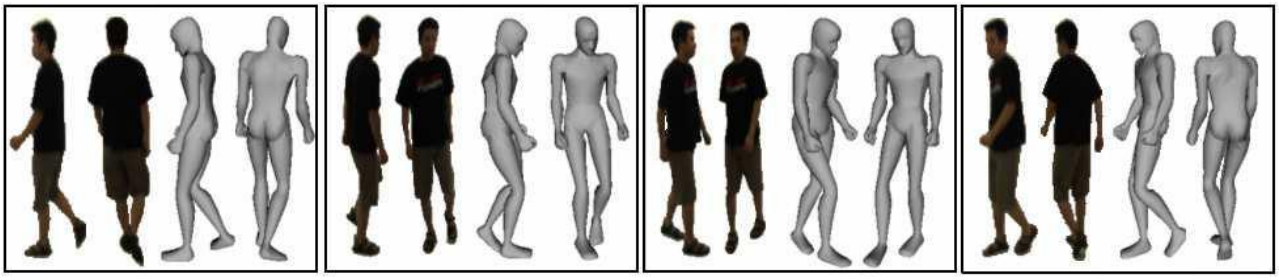


Рисунок 3.3 – Результати захоплення руху з реальних даних за допомогою 2 синхронізованих некаліброваних камер. Усі захоплені пози представлені в загальній моделі сітці і виражені з тих самих кутів, що і камери.

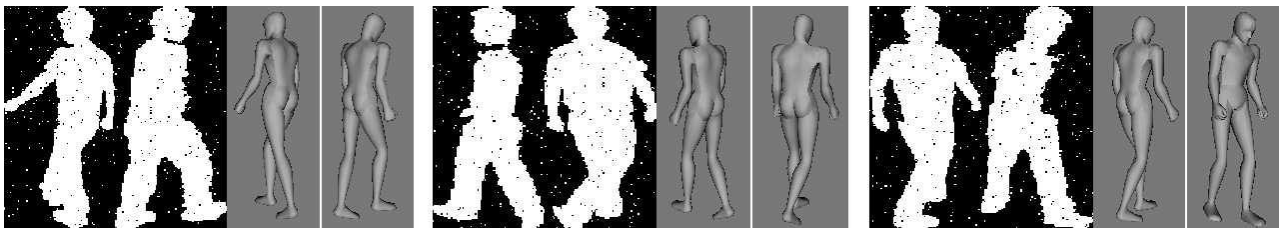


Рисунок 3.4 – Вибрані результати зйомки для ілюстрації надійності цієї техніки.

3.2 Тривимірне відстеження людського тіла в режимі реального часу за допомогою моделі Маркова з змінною довжиною

Як вже було зазначено, відстеження людей важке через високу розмірність кінематики повного тіла, швидких рухів та частоти самооклюзій. Більш того, вільний одяг, тіні або шум фотокамери можуть ще більше ускладнити проблему виводу. Один із підходів до відстеження як глобальної задачі оптимізації - почати з даних зображення, намагаючись самостійно виявляти функції в кожному кадрі.

Простір параметрів, доповнений похідними першого порядку, автоматично розподіляється на гауссові кластери, кожен з яких являє собою елементарний рух: поширення гіпотез всередині кожного кластера є точнішим та ефективним. Переходи між кластерами використовують передбачення моделі Маркова з змінною довжиною, що може пояснити поведінку високого рівня протягом тривалого часу.

Використання методів Монте-Карло [23], оцінка кандидатів моделі має вирішальне значення для швидкості і надійності. У цьому методі представлено нову схему оцінювання, яка базується на об’ємній реконструкції та встановленні крапель, де зовнішній вигляд моделі та зображення свідчать про гаусовскі суміші.

Основним слабким місцем продуктивності при використанні методів Монте-Карло є оцінка функції правдоподібності. Для кожної частинки це, як правило, включає в себе генерацію 3D зовнішнього вигляду моделі зі стану частинки, продемонструвавши вигляд цієї моделі на доступній площині зображень, і порівнюючи її з деякими витягнутими ознаками зображеннями, такими як силуети або ребра.

3.2.1 Представлення людського тіла

Розглянемо принципи параметризації, яка використана для тіла людини, а також функції, які використовуються для вивчення моделі поведінки людини, що обмежуватиме пошук в рамках базисної системи байєсів [24].

Модель людського тіла базується на кінематичному дереві, що складається з 14 сегментів, як видно на рис. 3.5. Кожна поза представлена 25-мірним вектором C_t, який складається з кутів суглобів, а також положення та орієнтації кореня кінематичного дерева.

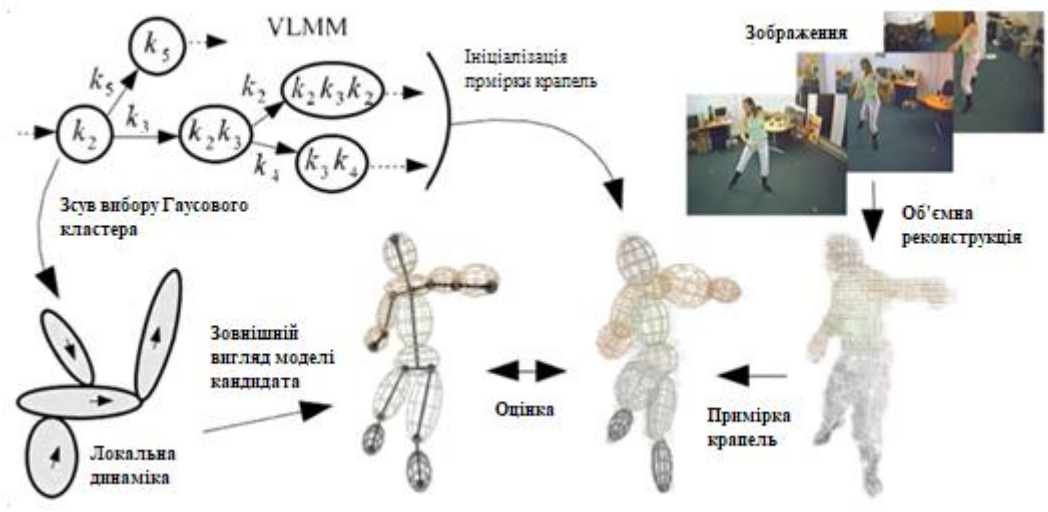


Рисунок 3.5 – Огляд системи

3.2.2 Кінематичне дерево та обмеження

Обмеження розміщуються на спільних обертаннях (виражених у кутах Ейлера) у вигляді обмежувальних значень. Надлишкові конфігурації та особливості усуваються, обмежуючи кожен суглоб до двох ступенів свободи. Обмеження скорочують кількість неможливих поз, але недостатньо, щоб відобразити складність морфологічних обмежень людини.

3.2.3 Представлення функціонального простору

Щоб навчитися лаконічній імовірнісній моделі руху 3D людини, потрібно вибрати відповідний простір функцій. Для кожної позиції тіла визначаємо відповідний вектор ознак $X_t = (x_t, \dot{x}_t)$ що складається з вектора суглобових кутів x_t та його першої похідної $\dot{x}_t$.

Поведінка людського тіла може розглядатися як гладка траєкторія всередині функціонального простору, яка відбирається за частотою кадрів, утворюючи послідовність векторних ознак X_t. Кожна послідовність описує

часову еволюцію пози тіла людини (доповненої першою похідною з суглобових кутів): $\{X_1, X_2, \ldots, X_m\}$.

3.2.4 Вивчення динаміки

Кластеризація функціонального простору. Через складність людської динаміки комплексну поведінку розбивають на елементарні рухи, для яких легше вивести місцеві динамічні моделі. Проблема полягає в тому, щоб автоматично ізолювати та моделювати ці елементарні рухи з навчальних даних.

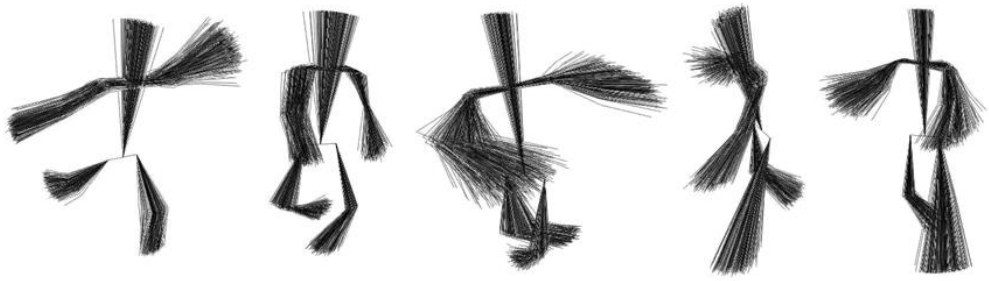


Рисунок 3.6 – Модельні контури, відібрані з різних гауссовских кластерів

Цього можна досягти об’єднавши функціональний простір в гауссові кластери, використовуючи варіант алгоритму EM [25]. Контури тіла, взяті з декількох кластерів на балетні дані, показані на рис. 3.6.

*Вивчення поведінки високого рівня з VLMM***.** Складні дії людини, такі як танці (або навіть більш прості, такі як ходьба), можна розглядати як послідовність примітивних рухів з структурою високого рівня, що контролює часове упорядкування.

Включивши імовірнісні знання основної поведінкової структури, як знімаються частки (в байєсівській системі відстеження з використанням моделювання Монте-Карло), можна розповсюджувати частки лише в правдоподібних напрямках, а також забезпечити автоматичні переходи між

різними моделями обумовень. Підходящим способом отримання такого знання є моделі Маркова зі змінною довжиною (VLMM) [26].

Змінна довжина марковських моделей стосується класу випадкових процесів, в яких тривалість пам'яті коливається, на відміну від n-го порядку марковських моделей. Їх перевага над схемою пам'яті марківської моделі - це їх здатність локально оптимізувати довжину пам'яті, необхідну для прогнозування. Це призводить до більш прозорого та ефективного представлення, яке особливо привабливе в тих випадках, коли потрібно зафіксувати часові залежності вищих порядків в деяких частинах поведінки та інших рівнях нижчого рівня.

VLMM можна розглядати як імовірнісний кінцевий автомат стану (PFSA) $\mathcal{M} = (Q, \Sigma, \tau, \gamma, s)$, де Σ це набір знаків, які представляють нульовий алфавіт VLMM, і Q - кінцевий набір модельних станів. Кожний стан відповідає рядку з довжиною Σ довжини не більше $N_{\mathcal{M}}(N_{\mathcal{M}} \geq 0)$, що представляє пам'ять для умовного переходу VLMM. Функція переходу τ, функція ймовірності виходу γ для певного стану та розподіл ймовірності Σ за початковими станами визначається як:

$$\tau: Q \times \Sigma \to Q \qquad \gamma: Q \times \Sigma \to [0,1] \quad s: Q \to [0,1]$$

VLMM - це генеративна імовірнісна модель: шляхом переміщення автомата моделі $\mathcal{M}$ можна генерувати послідовності знаків в Σ. Використовуючи набір гауссових кластерів як алфавіт, є можливість фіксувати тимчасові упорядкування та обмеження простору, пов'язані з примітивними рухами. Отже, перетинання $\mathcal{M}$ буде генерувати статистично правдоподібні приклади поведінки.

3.2.5 Прогнози, що використовують динамічну модель

Використовуючи правило Байєса, імовірність моделювання конфігурації x_t заданого виміру z_t становить:

$$P(x_t|Z_t) = k.P(z_t|x_t).\int P(x_t|x_{t-1}).P(x_{t-1}|Z_{t-1})dx_{t-1} \quad (3.6)$$

$P(x_t|Z_t)$ – задній,

$P(z_t|x_t)$ – умовна імовірність,

$P(x_t|x_{t-1})$ – рух попередній ,

$P(x_{t-1}|Z_{t-1})$ – попередній задній,

де k - нормалізуюча константа, і $Z_t = \{z_1, z_2, \dots, z_t\}$. Задній розподіл аппроксиміруется набором дискретних часток, кожна з яких являє собою конфігурацію тіла.

*Переходи між кластерами з VLMM***.** Частинки доповнюються своїм поточним станом VLMM q_t, з якого кластер k_t, до якого вони належать, легко виводиться. Переходи (або стрибки) між кластерами залежать від особливості вектора X_t частинки, а також від імовірності переходу γ в VLMM. Імовірність переходу до нового гауссовського кластера k_{t+1} середнього $\mu_{k_{t+1}}$ та коваріації $\Sigma_{k_{t+1}}$становить:

$$P(k_{t+1}|X_t, q_t) \infty P(X_t|k_{t+1}) \cdot P(k_{t+1}|q_t) =$$

$$= \frac{1}{\sqrt{(2\pi)^d|\Sigma_{k_{t+1}}|}}.e^{-\frac{1}{2}\left(X_t-\mu_{k_{t+1}}\right)^T\cdot\Sigma_{k_{t+1}}^{-1}\cdot\left(X_t-\mu_{k_{t+1}}\right)} \cdot \gamma(q_t, k_{t+1}) \quad (3.7)$$

На кожному кадрі перерахування стану вибирається у відповідності з вищенаведеними ймовірностями для кожного сусіднього кластера. На практиці, лише кілька переходів кодуються в VLMM, що робить оцінку ефективною. Якщо вибрано той самий кластер ($k_{t+1} = k_t$), частинка поширюється використовуючи локальну динаміку. Якщо вибрано новий кластер, параметри частинки повторно відібрані з нового гаусового кластера.

Локальна динаміка. Всередині кожного гаусового кластера нову конфігурацію моделі можна передбачити стохастично з попереднього вектора особливостей X_t. Оскільки гауссові кластери включають похідні, прогноз ефективно поводить себе як модель другого порядку. Розглянемо гауссовський

кластер середнього $\mu = \begin{pmatrix}\mu_x \\ \mu_{\dot{x}}\end{pmatrix}$ і коваріаційну матрицю $\Sigma = \begin{pmatrix}\Sigma_{xx} & \Sigma_{x\dot{x}} \\ \Sigma_{x\dot{x}}^T & \Sigma_{\dot{x}\dot{x}}\end{pmatrix}$. Вектор шуму безпосередньо відбирається з матриці коваріації кластера з коефіцієнтом ослаблення λ, що приводить до формулювання:

$$\begin{aligned} \dot{x}_t &= \dot{x}_{t-1} + \lambda . d\dot{x}_t \\ x_t &= x_{t-1} + \dot{x}_t + \lambda . dx_t \end{aligned} \qquad \begin{pmatrix} dx_t \\ d\dot{x}_t \end{pmatrix} \sim \mathcal{N}(0, \Sigma) \tag{3.8}$$

Випадковий вектор шуму зображується як $(dx_t\, d\dot{x}_t)^T = \sqrt{\Sigma} \cdot X$ з $X \sim \mathcal{N}(0,1)$. Квадратний корінь матриці коваріації обчислюється шляхом виконання розкладання на власні значеня $\Sigma = V \cdot D \cdot V^T$, і беручи квадратний корінь власних значень на діагоналі D , так що $\sqrt{\Sigma} = V\sqrt{D} \cdot V^T$.

Цю прогнозовану модель слід розуміти в контексті вибірки Монте-Карло, де шум вводиться для моделювання невизначеності в прогнозуванні: тому властивості шумового вектора настільки ж важливі, як сама динаміка. Матриця коваріації поточного кластера забезпечує гарне наближення цієї невизначеності, а вибірка шумового вектора з кластера сама по собі робить поширення невизначеності значно ближче до навчальних даних, ніж рівномірний гаусівський шум.

Щоб зберегти модель поведінки незалежно від глобальної позиції та орієнтації предмета, шість глобальних параметрів не моделюються гаусовськими кластерами і, отже, поширюються з рівномірним шумом.

3.2.6 Швидка оцінка ймовірності

Зовнішній вигляд моделі. Зовнішній вигляд моделюється 3D-краплями, прикріпленими уздовж кісток кінематичної моделі.. Форма згустку описується гаусовським розподілом середнього μ_X і коваріаційною матриці матрицею Σ_X.

Оскільки краплі утворюються в локальній системі координат кожної частини тіла, зберігають лише чотири вільних параметра: одиничне значення зміщення, яке підсумовує середню величину μ_X вздовж першої осі кістки, на якій прикріплена крапя, і три власних значення, які повністю описують коваріаційну матрицю Σ_X. Трансформація, необхідна для перетворення крапель з локальних на глобальні координати, отримується за допомогою прямої кінематики.

Краплі також включають інформацію про кольори, які, подібно до форми, яка представлена гауссовським розподілом середнього μ_C і коваріаційною матрицею Σ_C.

Оскільки колір кожної краплі унімодальний, одяг з кількома кольорами повинен бути оброблений сумішшю краплі. Починаючи з одного блоку для кожної частини тіла, процес "розщеплення та злиття" забезпечує оптимальний опис даних. Критерій, який використовується для вирішення, чи повинна бути розділена крапля - це колірна дисперсія вздовж основної просторової осі краплі. Це вимірювання отримано шляхом проектування змішаної коваріаційної матриці між просторовою та кольоровою інформацією Σ_{XC} (обчислених за даними з ЕМ) на напрямок поточної кістки у кінематичній моделі.

Об'ємна реконструкція. Об'ємна реконструкція має перевагу, поєднуючи відповідну інформацію для відстеження (форма та колір) у єдину когерентну структуру. Незважаючи на те, що інші функції, такі як ребра або текстура, можуть надавати цінну інформацію, неминуче розмиття руху заважає їх надійності при роботі з швидкими рухами. Алгоритми форма-від-силуету, використовуючи відповідності між переглядами камери, можуть забезпечити більшу стійкість та продуктивність, ніж вилучення окремих зображень на основі виділення ознак. У цьому методі, використовуючи калібровані камери, алгоритм візуальної оболонки проектує тривимірні вокселі на доступні площини зображень і зберігає ті, які лежать всередині всіх силуетів досліджуваного об'єкта. Об'єднання видобутку силуету та об'ємної реконструкції в ієрархічну схему, має подвійну перевагу: надійна статистика

пікселів та покращена продуктивність. Інформація про кольори також відновлюється, що робить реконструйований об’єм цінною базою для відстеження.

Оцінка часток. Конфігурація моделі (частинки) оцінюється спочатку генеруючи модель зовнішнього вигляду від стану частинки, а потім порівнюючи отримані краплі з результатами, отриманими з даних зображення. $F = \Sigma_i \alpha_i f_i$ суміш, отримана з моделі $G = \Sigma_i \beta_i \mathrm{g}_i$ і відповідає зображенням. Розбіжність Кулбака-Лейблера (KL) може бути використана для вимірювання крос-ентропії між двома сумішами:

$$D_{KL}(F||G) = \int F \ln \frac{F}{G} = \sum_i \alpha_i \cdot \int f_i \ln F - \sum_i \alpha_i \cdot \int f_i \ln G \tag{3.9}$$

Використовуючи наближення, для непересічних кластерів:

$$D_{KL}(F||G) = \sum_i \alpha_i \cdot \int f_i \ln \alpha_i \cdot f_i - \sum_i \alpha_i \max_j \int f_i \ln \beta_i \mathrm{g}_i =$$

$$= \sum_i \alpha_i \min_j \left(D_{kL}(f_i||\mathrm{g}_i) + \ln \frac{\alpha_i}{\beta i}\right) \tag{3.10}$$

Відповідність між краплями зберігається за формою $f_i \leftrightarrow \mathrm{g}_{\pi(i)}$, так що компромісна функція оцінки часу виконання є лінійною щодо кількості крапель:

$$D_{KL}(F||G) = \sum_{i=1}^{n} \alpha_i \left(D_{KL}\left(f_i||\mathrm{g}_{\pi(i)} + \ln \frac{\alpha_i}{\beta \pi(i)}\right)\right. \tag{3.11}$$

Останній вираз можна ефективно обчислити, використовуючи рішення замкнутої форми дивергенції KL між двома гауссовскими краплями $f \sim \mathcal{N}(\mu_f, \Sigma_f)$ та $\mathrm{g} \sim \mathcal{N}(\mu_\mathrm{g}, \Sigma_\mathrm{g})$:

$$D_{KL}(f||\mathrm{g}) = \frac{1}{2}(\ln\frac{|\Sigma_f|}{|\Sigma_\mathrm{g}|} - d + tr(\sum_f^{-1} \quad \Sigma_\mathrm{g}) + (\mu_\mathrm{g} - \mu_f)^T \sum_f^{-1} (\mu_\mathrm{g} - \mu_f)) \quad (3.12)$$

де d - розмірність гауссових крапель f та g .

Результати роботи моделі. Об'ємна реконструкція базується на 4 фотоапаратах, які знімають зображення на швидкості 30 кадрів в секунду з роздільною здатністю 320×240.

Тренувальні дані складалися з 8 послідовностей захоплення руху танцювальних рухів балету, приблизно 2000 кадрів кожен. При розподілі параметричного простору оптимальна кількість кластерів автоматично виявилася рівною 256, що може здатися досить високим значенням, але насправді відображає основну складність рухів. Випадкові збої відстеження, через рухи, які невидно у тренувальному наборі, виявляються та швидко відновлюються.

Повна система, включає в себе зйомку зображень, об'ємну реконструкцію та байєсову систему відстеження, працює на частоті 10 кадрів в секунду з наповненням 1000 частинок на одному комп'ютері з частотою 2 ГГц.

3.3 Відновлення пози тіла на основі 3D-скелету

Далі розглянуто підхід до відновлення рухів тіла з різних видів за допомогою 3D скелетної моделі. В якості вхідних значень послідовності силуетів на першому плані беруть з численних точок огляду та обчислюють для кожного кадру скелетну позу, яка найкраще підходить для тіла. Скелетні моделі кодують в основному інформацію про рух, і тому дозволяють відокремити оцінку руху від оцінки форми, для якої існують рішення. А фокусування на параметрах руху суттєво знижує залежність від конкретних форм тіла, даючи тим самим більш гнучкі рішення для захоплення руху тіла (рис.3.7).

Скелетна модель не містить жодної об'ємної інформації. Отже, вона має меншу залежність від розмірів тіла. Крім того, довжини кінцівок, як правило,

слідують біологічним природним законам, тоді як форми людей різняться серед населення.

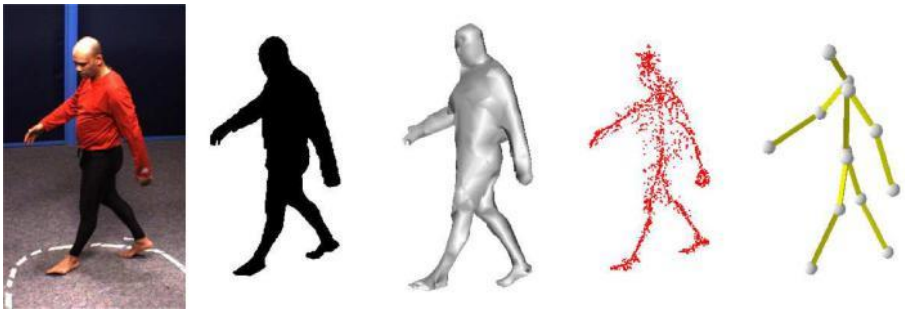


Рисунок 3.7 – Відстеження руху: (a) кольорове зображення; (b) силует; (c) візуальний корпус; (d) серединні точки осі (e); (f) скелетна поза.

3.3.1 Скелетна шарнірна модель

Розглянемо апріорно зчленовану модель, що представляє пози тіла (рис.3.8). Пропоновано використовувати зчленовану модель, не враховуючи будь-яку об’ємну інформацію про користувача.

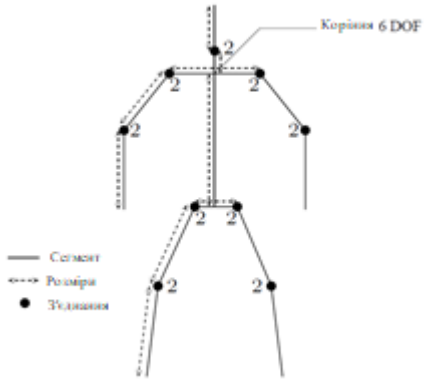


Рисунок 3.8 – Скелетна зчленована модель.

Ця скелетна зчленована модель складається з кінематичного ланцюжка сегментів. Оскільки інтерактивні програми зазвичай цікавляться принциповими суглобами (ліктями, плечами, колінами, ногами та головою), даний метод обмежує модель до 12 сегментів із 9 суглобами (рис. 3.9). Це призводить до 24

ступенів свободи: 2 на суглоби і 6 для позиції та орієнтації кореня. Інші моделі, що мають вищу вірність анатомії людини, можуть також використовуватися, якщо це вимагається більш вимогливими програмами (наприклад, графічними анімаціями). Для суглобів з 2 ступенями свободи було обрано представлення, засноване на кутах Ейлера. Щоб уникнути класичних нестаціонарних завдань, що виникають при ейлеровских параметризаціях, встановлено осі обертання у найбільш неправдоподібному напрямку (наприклад, через природні обмеження суглобів).

3.3.2 Спостережувані скелетні дані

Іншим важливим елементом процесу відстеження є дані, які розглядаються як вимірювання для пози тіла, до якої модель підходить. Відсутність точності в екстракції скелету компенсується апріорними знаннями (людською артикульованою моделлю).

У даному підході припускається, що доступні силуети, витягнуті з каліброваних камер з різними точками огляду. Ці силуети отримують за допомогою методів віднімання від фону. З цих силуетів спочатку обчислюється їх еквівалент 3D, тобто візуальна оболонка [27]. Для цього був використаний метод, який обчислює багатогранник у просторі. Ця форма точно реалізується на силуети у зображеннях і, таким чином, попередньо обслуговує всю інформацію про силует. Це потім обробляється, щоб витягти його внутрішню структуру, а саме про скелет. Цей крок, званий скелетонізацією, отримав помітну увагу від обчислювальної геометричної комунікації. Для скелетів можна розглянути кілька визначень, але найбільш успішним є, безумовно, медіальна вісь [28]. Медійна вісь визначаються як геометричне місце центрів замкнутих куль, які є максимальними щодо включення. У випадку дискретної поверхні процес, що веде до дискретного наближення медіальної осі, іноді називається "Медіальна трансформація осей" (Medial Axis Transform).

Важливим недоліком дискретної медіальної осі є його чутливість до шуму (рис. 3.9 (b)). Однак є алгоритми, які враховують шум форми вхідних даних [29]. Ідея полягає в тому, щоб спершу обчислити дискретну медіальну вісь, а потім - обрізати її, щоб усунути зайві елементи. Алгоритм переходить потім в такий спосіб:

1. Центри Вороного обчислюються з сітки вершин. Розглядаються лише центри, що лежать всередині сітки (рис. 3.9 (b)).

2. Для кожного центру C одержуємо відповідний тетраедр Делоне (P_1; P_2; P_3; P_4) і обчислюють: його радіус $\rho(C)= d(C,P_1)$, його бісектрису $\theta(C)= \max_{i\neq j} (\widehat{P_iCP_j})$. Зайві елементи виключаються на основі мінімального радіусу та порогового кута бісектриси.

Це призводить до набору 3D точок $\{X_0 \dots X_n\}$, що ми названі "дані скелету" (рис. 3.9 (c) та (d)). Вибір радіуса та похилу кутів бісектриси полягає в знаходженні компромісу між якістю скелета та кількістю отриманих точок. Дійсно, чим вище порогові значення, тим краще скелет, проте менше пунктів вибираються (рис. 3.9 (d)). На практиці встановлюється порогове значення радіусу 4 см, а поріг двовимірного кута близько 160 (рис. 3.9 (c)). Слід зазначити, що 3D-медіальна ось не є кривою, як у 2D, а поверхнею.

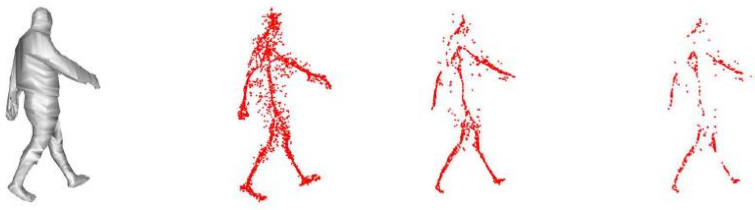


Рисунок 3.9 – (a) Отримана точна візуальна оболонка. (b) Внутрішні центри, що породжують шумний скелет. (c) Скелетонізація після обрізання при r > 4 см θ > 160^o : більшість зайвих елементів вилучено . (d) Скелетонізація після обрізки r > 5см та θ > 170^o

3.3.3 Генеративна модель

Для того, щоб отримати пози користувача за заданий час t, потрібно визначити зв'язок між апріорною зчленованою моделлю та спостереженими даними. Першим рішенням була б характеристика подібності між набором точок скелета $\{X_0 \dots X_n\}$, скелетною моделлю S, яка базується на відстані кожної точки до найближчого зчленованого сегмента $s \in S$, як в такому випадку:

$$P(\{X_i\} \quad S) = P(S) \times \prod_{i=0}^{n} P(X_i|S), \tag{3.13}$$

де $P(X_i|S) = N(d(X_i,S),\sigma^2)$ та $d(X_i,S) = \min_{s\in S}(X_i,S)$, з $d(X_i,S)$, що представляє евклідову відстань.

Проте, максимізація відповідного поперечного розподілу $P(S|\{X_i\})$ призводить до труднощів. Дійсно, прикріплення точки до сегменту може змінюватися під час монтажу, створюючи невідповідності та градієнтні розриви. Щоб вирішити цю проблему, вводимо приховані змінні a_i, по одному для кожної точки, що представляє сегмент, прикріплений до точки X_i. Тоді спільна ймовірність спостережуваних даних і пози: $P(\{X_i\}\{a_i\}S) = P(S) \times \prod_{i=0}^{n} P(a_i|S) \times \prod_{i=0}^{n} P(X_i|a_iS)$, де:$P(S)$ є попереднім розподілом пози. $P(a_i = j|S)$ являє собою апріорне значення на вкладенні з єдиним знанням пози. Його встановлено пропорційно довжині відповідного сегмента s_j.

P ($X_i|a_i = jS$) - представляє ймовірність того, що точка Xi належить до кінцівки, що відповідає сегменту s_j.

Знаходження найкращої пози полягає в тому, щоб максимізувати нащадків:

$$\begin{aligned} P(S|\{X_i\}) &\propto \Sigma\{a_i\}P(\{X_i\}\{a_i\}S), \\ &\propto \prod_{i=0}^{n} \Sigma a_i P(X_i a_i S), \\ &\propto P(S)\prod_{i=0}^{n} \Sigma a_i P(a_i|S)(X_i|a_i S). \end{aligned} \tag{3.14}$$

На відміну від першого рішення (3.13), цей нащадок добре адаптований для максимізації, оскільки всі його похідні є безперервними (функція C∞). Цей нащадок також є більш надійним, оскільки він маргіналізує всю можливу точку сегментації вкладень замість розгляду єдиного можливого приєднання з точки до найближчого сегмента.

Для того, щоб знайти максимальну апостеріорну оцінку MAP (maximum a posteriori estimate) та як класичну при роботі з прихованими змінами, використовується підхід максимізації очікувань, де

Крок E полягає у розрахунку умов очікування $E(a_i=j)$ для поточної оціночної пози $\bar{S}$:

$$\begin{aligned} E(a_i=j) &= P(a_i = j|X_0 \dots X_n\bar{S}), \\ &= P(a_i = j|X_i\bar{S}), \\ &= \frac{P(a_i=jX_i\bar{S})}{\Sigma_{a_i}P(a_i\,X_i\bar{S})}; \end{aligned} \tag{3.15}$$

Крок M полягає у знаходженні максимального значення пози S: $F(S)=\sum_{i=0}^{n}\sum_{a_i}E(a_i)\times logP(a_iX_iS)$, $P(a_iX_iS) = P(S)(a_i|S)(X_i|a_iS)$.

$P(S)$, вважається рівномірним, а попередній розподіл на a_i не залежить від пози.

Це призводить до максимізації: $F(S)\infty\sum_{i=0}^{n}\sum_{a_i}E(a_i)\times logP(X_i|a_iS)$. Це еквівалентно мінімізації його відхиленого вигляду:

$$\sum_{i=0}^{n}\sum_{a_i=j}E(a_i=j)\times\frac{d(X_i,s_j)^2}{2\sigma_j^2}.$$

Ця формула визначає проблему найменших квадратів. Використовується добре відомий алгоритм мінімізації Левенберга-Марквардта, оскільки він добре пристосований до цього типу проблем.

3.3.4 Відстеження руху користувача

Процес приєднання відновлює одну позицію на заданому кадрі. Щоб відновити рух користувача, потрібно описати, як отримується поза s_{t+1} на кадрі $t+1$ знаючи попередні пози. Ця проблема «поширення» полягає в прогнозі ймовірного положення s'_{t+1}. Це передбачення використовується в якості початкового наближення в процесі мінімізації, в результаті чого остаточна поза s_{t+1}. Це передбачення зазвичай базується на динамічній моделі, такій як постійна швидкість або постійне прискорення. Ці моделі ефективні при моделюванні витіснення об'єктів з відносно стійкою швидкістю. Ця умова зазвичай передбачає невелике співвідношення між застосованими силами та масою об'єкта. Якщо цей стан дійсний для кореневого положення та орієнтації тіла, це явно непридатне для рук або ніг. Їхні рухи можуть бути дуже динамічними. У таких випадках відстеження без динамічної моделі ($s'_{t+1} = S_t$) є гарним рішенням.

Результати методу відстеження тіла. Метод відстеження тіла, представлений у попередніх секціях, був впроваджений та випробуваний на різних послідовностях природних рухів, таких як ходьба в будь-якому напрямку.

Послідовності зображень були отримані з використанням 6 Firewire камер зйомки 780×580 зображень на 27 кадрів в секунду. Ці фотоапарати запускаються в електронному режимі для забезпечення синхронізації між зображеннями. Силуєти отримують за допомогою стандартного методу віднімання фону.

Результати відстеження послідовності ходьби представлені на рис. 3.10. Перевірка здійснюється візуально, перевіряючи кожен кадр. Тільки 6 кадрів із 400 були виявлені частково невірно відстежуваними. Ці 6 кадрів організовані у 2 групах з 3 послідовних кадрів, 2 групи, що відповідають такої ж ситуації в послідовності, але в різний час.

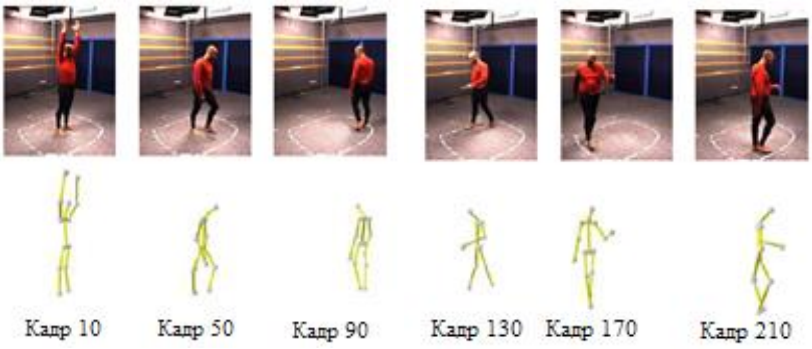


Рисунок 3.10 – Скелет створений в різний час для послідовності ходьби.

У цьому положенні ліктьовий суглоб був виявлений невірно від його реального положення. Ця ситуація пов’язана з проблемами видимості, які призводять до помилок скелетних даних між тулубом і рукою.

Послідовності не виконуються в певних середовищах, таких як сині кімнати, в результаті чого шумні силуети, отримані шляхом віднімання фону (див. рис.3.11).

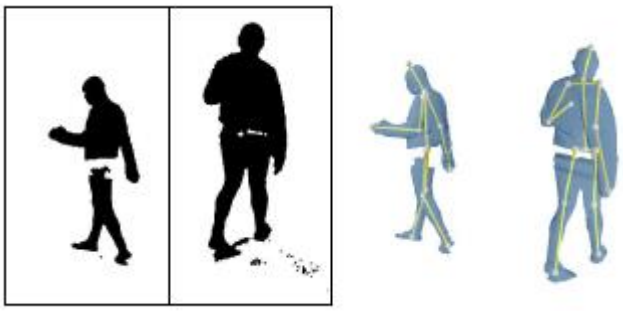


Рисунок 3.11 – Ліворуч, приклади шумних силуетів у послідовності. Праворуч, результат оцінювання скелетної пози із цими силуетами (з 2 різних точок зору).

Обчислення скелетних даних: Візуальне обчислення корпусу може бути досягнуте в реальному часі з затримкою 70 мс. Складність скелетнізації полягає, по суті, в обчисленні клітин Вороного. Це займає близько 60 мс для 2000 поверхневих точок на Opteron 2 ГГц. Розподіл його обчислень дозволяє

цей процес запускати зі швидкістю 30 кадрів на секунду, але не зменшує його прихований стан. Потужність в режимі реального часу - менш ніж 30 мсек. Відстеження займає близько секунди на кадр. Велику частину часу займають обчислювальні відстані від точок до сегментів моделі.

Висновки до розділу

У цьому розділі описано три безмаркерні системи захоплення руху людини. Наведено опис методів, детально описані алгоритми засновані на оцінці ймовірності, за якими відбуваються відстеження моделі.

Метод описаний у пункті 3.1 забезпечує хороші результати, якщо топологія реконструйованої 3D-форми відповідає топології людини, тобто кожна сторона тіла однозначно відображається на оцінку тривимірної форми. З випадками самооклюзії або великими контактами між кінцівками та тілом метод часто не працює належним чином.

Метод, представлений у пункті 3.2 включає в себе форму та кольорову підказку. Такий метод вимагає використання контрастного одягу між кожною частиною тіла для коректного відстеження, тим самим додавши обмеження зручності користування. Цей метод забезпечує захоплення руху в режимі реального часу також з декількох видів та працює із частотою інтеркактивних кадрів (10 кадрів в секунду).

Для методу описаному в розділі 3.3 необхідним є ручне втручання для антропометричних вимірювань та оцінки початкової позиції.

Кожна модель не є ідеальною та містить велику кількість складних розрахунків. Ці методи не є достатньо зручними для користувача.

4 ДОСЛІДЖЕННЯ БЕЗМАРКЕРНОЇ СИСТЕМИ ТРИВИМІРНОГО ЗАХОПЛЕННЯ РУХУ ЛЮДИНИ З ВИКОРИСТАННЯМ КІЛЬКОХ ВИДІВ З КАМЕР

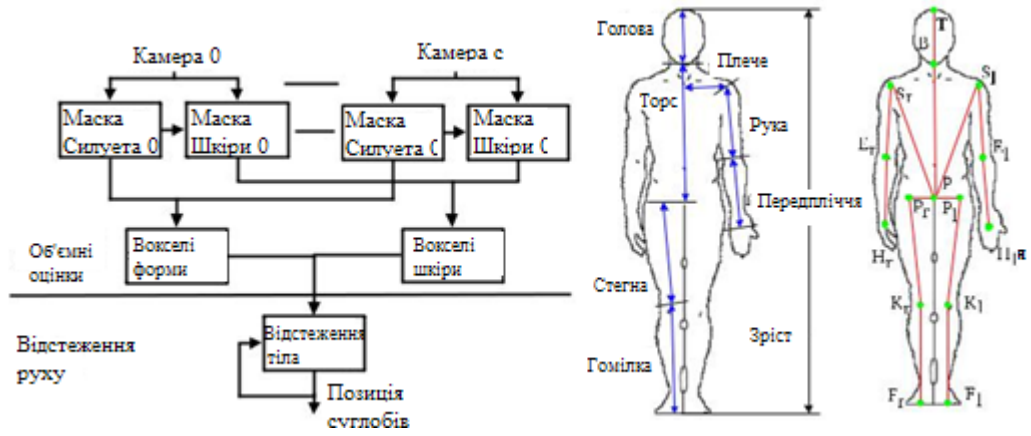
В даній роботі для створення повністю автоматизованої системи тривимірного захоплення руху людини в режимі реального часу без маркерів, було досліджено функціональні можливості і особливості системи основаної на апаратному рішенні, з використанням декількох камер.

Підхід, заснований на швидких алгоритмах, використовує прості методи і вимагає недорогих пристроїв. Використовуючи вхід з декількох еталонних веб-камер, розширений алгоритм Shape-From-Silhouette реконструює рух людини в режимі реального часу. Завдяки швидким та простим алгоритмам і недорогим камерам система ідеально підходить для домашніх розважальних пристроїв та навчальних потреб.

4.1 Огляд методу

У попередньому розділі було проаналізовано реалізацію кількох методів для вирішення проблеми захоплення руху без маркера. Вони різняться в залежності від характеристик, використовуваних для аналізу, і кількості та функціональності задіяних камер.

Метод [30], що пропонується у цьому розділі має повністю автоматизовану систему для практичного захоплення руху в реальному часі завдяки використанню декількох камер. Він включає в себе етап ініціалізації та відстеження руху (рис.4.1). Процес базується на простих прийомах, заснованих на аналізі топології форми і шкіри. Він працює зі швидкістю 30 кадрів в секунду на одному стандартному комп’ютері. При цьому не вимагається паралельних інтенсивних обчислень або пакетної обробки.



а) б) в)

Рисунок 4.1 – (а) Огляд системи: Алгоритм реконструкції та алгоритми оцінки положення. Маркування частин тіла (б) і спільне позначення (в).

4.2 Оцінка трьохмірної форми і шкіри

Основою запропонованого алгоритму є розширення алгоритмів Shape-From-Silhouette (SFS), що дозволяє реконструювати в реальному часі тривимірну форму та 3D кольорові частини людини з каліброваних камер. Зазвичай, тільки в методах SFS обчислюється оцінка об’єкта 3D-форми в реальному часі, від її силуетних зображень. Силует зображення є двійковими масками, відповідних захоплених зображень, де 0 відповідає фону, а 1 позначає особливість об’єкта.

За визначенням, об’єкт лежить всередині об’єму, який генерується шляхом зворотного проектування його силуету через центр камери (називається конус силуету). При одночасному перегляді з декількох об’єктів одного і того ж об’єкта на перехресті всіх конусів силуету створюється том, який називається "візуальний корпус", який гарантовано містить реальний об’єкт. Існує, в основному, два способи обчислення візуальної оболонки об’єкта.

Поверхісні підходи. В даному випадку обчислюють перетин конусних

поверхонь силуету (рис.4.2 (б)). Перші силуети перетворюються на багатокутники. Кожне ребро проектується назад, щоб сформувати 3D-полігон. Потім кожен 3D-полігон проектується на зображення один одного і перетинається з кожним силуетом в 2D. Отримані багатокутники зібрані, щоб сформувати оцінку багатогранної форми. Отримана поверхнева форма від силуету підкреслена на рис.4.2 (б). Крім того, неповні або пошкоджені поверхневі моделі можуть бути створені безпосередньо в залежності від багатогранника, різкості і шуму силуету.

а) б) в)

Рисунок 4.2 – (а) Налаштування взаємодії. (б) Об’єкт реконструкції поверхневих та об’ємних підходів, представлених у 2D. (в) Обчислення SFS з використанням методу "проектування текстур"

Об’ємні підходи. Для цього підходу оцінюємо форму, обробляючи набір вокселей. Область обробки об’єкта поділяється на тривимірну сітку вокселей (елементів обсягу). Кожен воксел залишається частиною оціночної форми, якщо проекція на все зображення лежить у всіх силуетах (див. рис. 4.2 (в)). Цей об’ємний підхід адаптований для оцінки пози в реальному часі через його швидке обчислення і стійкість до зашумленості у силуетах.

Пропоновано нову структуру, яка обчислює тривимірну об’ємну форму і оцінку частин шкіри на одному комп’ютері. Реалізація системи складається з

двох задач:

1. Отримання вхідних даних: дані калібрування камери, отримання силуету та сегментація деталей шкіри.

2. Оцінка 3D форми та компонентів шкіри: вокселі форми обчислюються за допомогою реалізації графічного процесора SFS, а деталі шкіри визначаються, використовуючи видимість вокселів. Кожне завдання описано у відповідному пункті.

4.2.1 Отримання вхідних даних

По-перше, веб-камери калібруються за допомогою алгоритму, запропонованого у [31]. Щоб забезпечити узгодженість між камерами, калібрування кольорів здійснюється за допомогою методу, запропонованого Н. Джоші [32].

Другий етап полягає в сегментації силуету [33]. Для цього використовується метод, запропонований [34]. Спочатку потрібно отримати зображення фону. Потім передній план (людину), виявляють в пікселях, значення яких змінилося. Припускаємо, що лише одна особа знаходиться в полі зору камер, таким чином, він або вона представлені лише одним компонентом connex (з’єднання, зв’язок точок і прямих на площині). Через шум вебкамери можна отримати декілька часточок коннекс, але найменші з них видаляються: вони відповідають шуму.

Останній крок до оцінки форми - вилучення частини шкіри з силуету та кольорових зображень. Для цього використано метод нормалізованого пошуку таблиці (Normalized Look-up Table) [35], що забезпечує швидку сегментацію кольору шкіри. Ця сегментація застосовується до кожного зображення, що обмежує його силует маскою (пікселі з кольором шкіри поза силуету відповідають фоновим пікселям).

4.2.2 Оцінка 3D-компонентів та компонентів шкіри

Для вирішення задачі оцінки 3D-компонентів та компонентів шкіри, спочатку потрібно оцінити тривимірну форму людини, захопленої за допомогою графічного процесора, що реалізує SFS. Об’ємний SFS зазвичай грунтуються на проекції вокселей: воксель залишається частиною оціночної форми, якщо вона проектує себе в кожен силует. Для знаходження найкращого способу для реалізації GPU, використовується взаємна власність (reciprocal property). Кожен силует проектується у 3D-воксельну сітку, як це запропоновано в [34]. Якщо воксель є перетином усіх виступів силуету, то він являє собою оригінальний об’єкт.

Класичний куб N3 вокселя можна розглядати як стек N зображень з роздільною здатністю N × N. У досліджені зображення N буде складено у паралельних площинах екрана. Силуетні маски проектуються на кожну окрему ділянку за допомогою методу проектування текстур [36]. Перетин проекцій силуетів на всіх зрізах забезпечує формування на основі вокселів 3D-форми. Перетин проекцій масок силуету на одному зрізі вказується на рис.4.2 (c). Щоб зберегти пропускну здатність відеопроцесора, обчислення для куба вокселей виконуються в одному кадровому буфері, який розбитий усіма N зрізами дозволу N × N.

Щоб оцінити вокселі шкіри, обчислюємо видимість кожного вокселя з кожної камери. Тест на видимість вокселя базується на методі «буфер предметів» (Item Buffer), який використовується в деяких алгоритмах розфарбування вокселів [37]. Унікальний ідентифікатор пов’язаний з кожним вокселем (наприклад, кольором), а вокселі відображаються на растрових кадрах, відповідних кожному виду камери. Для кожного фреймбуфера кольору описують видимі воксели, і це дозволяє двонаправленим пікселям зіставляти воксель. Якщо воксель узгоджений зі шкірою тобто він відображається до пікселів маски шкіри у всіх переглядах камери), то він класифікується як

воксель шкіри. Щоб поліпшити час обчислення видимості, тестуються лише поверхні вокселей (тобто вокселі, які мають менше 26 сусідів).

Щоб скоротити час обчислень для оцінки пози, пропонується зберегти видимі вокселі.

Нехай $V\overline{\boldsymbol{skin}}$ - вибрані вокселі, що утворюють набір вокселей форми, V_{skin} - послідовний воксельний набір, а V_{all} – їх обєднання.

Запропонована реалізація забезпечує до 100 реконструкцій в секунду. Оскільки збір даних веб-камери здійснюється при 30 кадрах в секунду, це дозволяє заощадити час для обчислення захоплення руху, отже, досягнення мети в реальному часі реально.

4.3 Захоплення руху

Мета захоплення руху - визначити позу тіла протягом всього часу. Можливо визначити позу тіла, якщо можливо пов’язати кожен воксель з частиною тіла. Маркування суглобів представлена на рис.4.3.

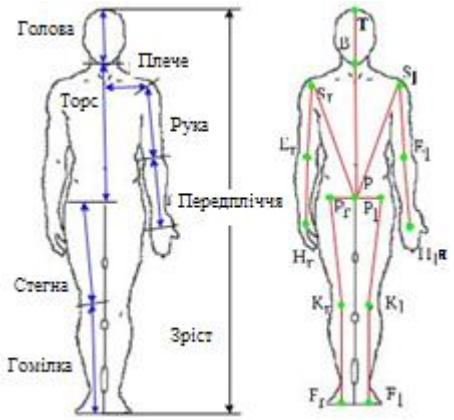


Рисунок 4.3 – Маркування суглобів

Для вирішення задачі захоплення руху запропоновано підхід, що є менш точний, ніж методи, засновані на реєстрації, але він дозволяє працювати в режимі реального часу. Надійність підвищується за рахунок використання

мультимодальної схеми, що складається з аналізу форми і шкіри, часового узгодження та антропометричних обмежень людини.

Представлена система працює на двох етапах: ініціалізація і відстеження; обидва використовують один і той же алгоритм з різними початковими умовами. Крок ініціалізації оцінює антропометричні значення і початкову позу. Потім, використовуючи цю інформацію, на останньому кроці відстежуються загальні положення частин тіла. Наші передумови полягають у тому, що обидві руки і обличчя частково розкриті, що тулуб у одязі, та одяг має колір відмінний від кольору шкіри.

Загальні позначення для подаліших викладок наступні:

Lx – довжина частини тіла x (див.рис. 4.1 (б));

D_x – орієнтація частини тіла;

R_x –радіус частини тіла (сфера або циліндр);

J^n – значення величини, що визначає спільне положення J, набір вокселей в кадрі n;

l та r позначають відповідно ліву і праву сторони;

V_x – позначає набір вокселей;

E_{Vx} – інерційний еліпсоїд набору вокселей;

$Cog\ (Vx)$ – центр тяжіння набору вокселей;

$Q(i)$ – позначає значення кількості Q на етапі i при роботі з ітеративними алгоритмами.

4.4 Відстеження частин тіла

Для відстеження частин тіла припускаємо, що попередня поза тіла і антропометричні оцінки відомі. Використовуючи 3D-оцінку форми і частини 3D-шкіри, відстежуємо частини тіла людини в реальному часі. Алгоритм відстеження працює з використанням активних вокселей Vact. Цей набір вокселів ініціалізується для всіх вокселів Vall і оновлюється на кожному кроці,

видаливши вокселі, використовувані для оцінки частин тіла. Тоді, оскільки тулуб з'єднаний з головою, відстежуємо тулуб. Далі, обчислюємо суглоби кінцівок, які пов'язані з тулубом.

*Відстеження голови***.** Цей крок спрямований на пошук T^n и B^n, відповідно положень верхньої частини голови і точки з'єднання між головою і шиєю в кадрі n. Головні артикуляції відслідковуються за допомогою алгоритму складання сфери [38]. Щоб прискорити процес, ініціалізуємо центр сфери в поточний центр обличчя, вилучений з безлічі вокселей шкіри.

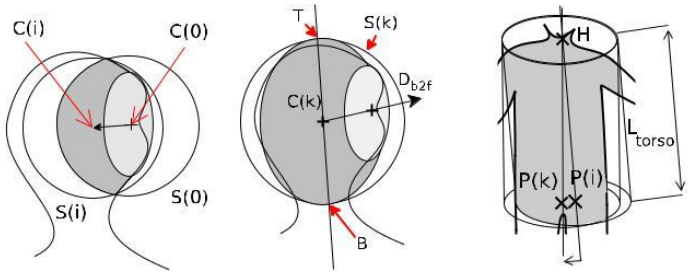
Нехай $\boldsymbol{V_{face}^{n}}$ - вокселі обличчя в поточному кадрі. $\boldsymbol{V_{skin}^{n}}$ містить вокселі обличчя і рук. Використовуючи часові критерії когерентності, $\boldsymbol{V_{face}^{n}}$ є найближчою (використовуючи точку-еліпсоїд евклидової відстані) компонентою зв'язку $\boldsymbol{V_{skin}^{n}}$ від попереднього набору лицьових вокселей $\boldsymbol{V_{face}^{n-1}}$.

Центр голови C^n обчислюється шляхом підгонки сфери *S(i)* в $\boldsymbol{V_{act}^{n}}$ (див. рис.4.4). Сфера *S(i)* визначається її центром $C^n(i)$ та радіусом R_{head}.

Алгоритм кріплення голови. $C^n(0)$ ініціалізується як центроїд $\boldsymbol{V_{face}^{n}}$. На кроці *i* алгоритму, $C^n(i)$ є центроїдом множини $\boldsymbol{V_{head}^{n}(i)}$ активних вокселей, які лежать в сфері *S(i-1),* яка визначається його центром $C^n(i-1)$ і його радіусом R_{head} (рис 4.4 (a)).

Алгоритм виконує ітерацію до етапу *k*, коли положення C^n стабілізується, тобто відстань між $C^n(k-1)$ і $C^n(k)$ падає нижче порога ϵ_{head}.

Оцінка суглобів. Знання C^n положення, B^n (відповідно T^n) обчислюється як нижній (верхній) перетин між *S(k)* і головною віссю голови $\mathcal{E}\boldsymbol{V_{head}^{n}}$ (рис. 4.4 (b)). Зворотній напрямок $\boldsymbol{D_{b2f}^{n}}$ визначається як напрямок від C^n до центру тяжіння $\boldsymbol{V_{face}^{n}}$ обличчя (при цьому вокселі з задньої частини голови не знаходяться в V_{skin}). На цьому етапі видаляємо з $\boldsymbol{V_{act}^{n}}$ дію множини елементів, що належать до голови $\boldsymbol{V_{head}^{n}}$.



(a) (b) (c)

Рисунок 4.4 – (a) Сферична установка (світло-сірий позначає $\boldsymbol{V^n_{face}}$, темно-сірий позначає $\boldsymbol{V^n_{head}(i)}$), (b) оцінка суглобів і (c) сегментація торса за допомогою установки циліндра.

Відстеження торса. Цей крок спрямований на визначення P^n положення таза шляхом установки циліндра в $\boldsymbol{V^n_{act}}$.

Оцінка форми тулуба циліндром забезпечує простий і швидкий метод локалізації таза. Нехай $\boldsymbol{V^n_{torso}}$- це множина вокселей, які описують торс, вони ініціалізуються з використанням вокселей $\boldsymbol{V^n_{act}}$. На кроці *i* алгоритм оцінює $\boldsymbol{D^n_{torso}}$, встановлюючи циліндр *CYL (i - 1)* в $\boldsymbol{V^n_{torso}(i)}$ (рис.4.4(c)). *CYL (i)* має ковпачок закріплений на B^n, як радіус R_{torso}, його довжина дорівнює L_{torso}, а його вісь - $\boldsymbol{D^n_{torso}(i)}$.

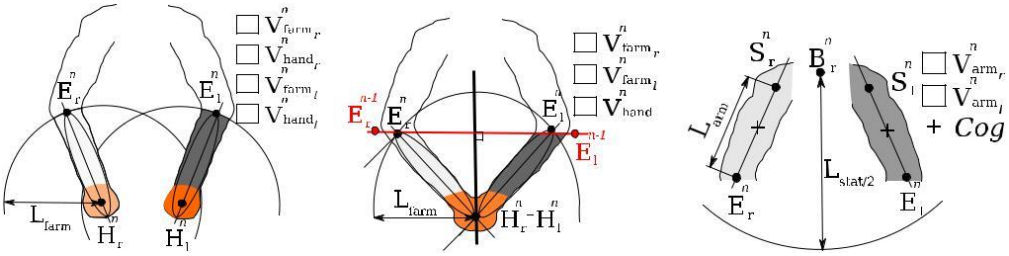
Алгоритм апроксимації торса. $\boldsymbol{V^n_{torso}(0)}$ ініціалізується $\boldsymbol{V^n_{act}}$, а вектор з B^n в P^{n-1} визначає початкове значення $\boldsymbol{D^n_{torso}(0)}$.

На етапі *i*, $\boldsymbol{V^n_{torso}(i)}$ обчислюється як множина елементів з $\boldsymbol{V^n_{torso}(i\text{-}1)}$, які лежать в *CYL(i - 1)*. Тоді $\boldsymbol{D^n_{torso}\ (i)}$ є головною віссю $\boldsymbol{\mathcal{E}V^n_{torso}\,(i)}$ (рис.4.4 (c)). Алгоритм виконує ітерацію до етапу *k*, коли відстань між віссю *CYL(k)* і центроїдом $\boldsymbol{V^n_{torso}(k)}$ падає нижче порога ϵ_{torso}. Положення P^n визначається як центр нижнього ковпачка *CYL (k).*

Глобальна орієнтація тіла. Орієнтація по вертикалі $\boldsymbol{D^n_{t2d}}$ отриманого суб'єкта задається як P^n-B^n. $\boldsymbol{D_{b2f}}$ був обчислений в розділі 4.4.1. Горизонтальна орієнтація $\boldsymbol{D^n_{l2r}}$ отриманого суб'єкта задається як $\boldsymbol{D^n_{l2r} = D^n_{t2d} \times D^n_{b2f}}$. Далі $\boldsymbol{V^n_{act}}$ оновлюється шляхом видалення елементів, що належать $\boldsymbol{V^n_{torso}}$.

Відстеження рук і передпліччя. Алгоритм для обчислення спільних положень передпліччя полягає у наступному: спочатку обчислюємо позиції рук з вокселей. Далі за допомогою антропометричного вимірювання довжини передпліччя визначаємо положення ліктів. Для обчислення їх сторін використовується часова когерентність.

Нехай $\boldsymbol{V_{hand}^{n}}$ - множина потенційних вокселей рук. L_{stat} / 2 - верхня границя довжини руки. $\boldsymbol{V_{hand}^{n}}$- визначається вокселями $\boldsymbol{V_{skin}^{n}}$- $\boldsymbol{V_{face}^{n}}$, які лежать всередині сфери, визначеної їх центром B^n і їх радіусом L_{stat} / 2. $\boldsymbol{V_{skin}^{n}}$ містить вокселі рук і обличчя. На рис.4.5 показані різні конфігурації передпліччя:



(a) (b) (c)

Рисунок 4.5 – Відстеження передпліч в різних положеннях рук: (a) дві різні конфігурації передпліч (b) конфігурація з'єднаних рук. (c) Ілюстрація алгоритму відстеження плечей.

Дві чітко видимі руки. $\boldsymbol{V_{hand}^{n}}$ містить кілька компонентів коннекс. Нехай $\boldsymbol{V_{hand0}^{n}}$ та $\boldsymbol{V_{hand1}^{n}}$ є двома найбільшими компонентами, відповідними двом рукам з $\boldsymbol{H_{x}^{n}}$ =Cog($\boldsymbol{V_{handx}^{n}}$) та $x \in [0, 1]$.

Передпліччя мають постійну довжину L_{farm} в часі. Потенційними вокселями для передпліччя x є воксели з $\boldsymbol{V_{act}^{n}}$, що лежать в сфері радіуса L_{farm}, центрованої в $\boldsymbol{H_{x}^{n}}$. Компонент connex цих вокселей, який містить $\boldsymbol{H_{x}^{n}}$, являє собою передпліччя x. Нехай $\boldsymbol{V_{farmx}^{n}}$ - це компонент коннекс; є два можливі випадки, щоб ідентифікувати лікоть.

Якщо передпліччя не стикалися, тобто $\boldsymbol{V_{farm0}^{n}} \cap \boldsymbol{V_{farm1}^{n}} = \emptyset$, то використовуємо основну вісь $\mathcal{E}\,\boldsymbol{V_{farm\,x}^{n}}$ та L_{farm} для обчислення положення ліктя

$\boldsymbol{E_x^n}$. Сторони обчислюються з використанням критеріїв часової когерентності: сторона передпліччя збігається з найближчим передпліччям, обчисленим в попередньому кадрі. Ця конфігурація підкреслена на рис.4.4(a). В іншому випадку, коли передпліччя стикаються і $\boldsymbol{V_{farm0}^n} \cap \boldsymbol{V_{farm1}^n} = \emptyset$ спочатку визначаємо сторони властивістю постійної довжини передпліччя. $\boldsymbol{H_x^n}$ є правостороннім, якщо

$$\left|\left|\boldsymbol{d}(\boldsymbol{H_x^n}, \boldsymbol{E_r^{n-1}}) - \boldsymbol{L_{farm}}\right|\right| < \left|\left|\boldsymbol{d}\left(\boldsymbol{H_x^n}, \boldsymbol{E_l^{n-1}}\right) - \boldsymbol{L_{farm}}\right|\right|, \qquad (4.1)$$

інакше $\boldsymbol{H_x^n}$ залишається лівостороннім.

Воксели v_i з $\boldsymbol{V_{farm0}^n} \cup \boldsymbol{V_{farm1}^n}$ сегментовані в дві частини $\boldsymbol{V_{farmr}^n}$ та $\boldsymbol{V_{farml}^n}$, використовуючи алгоритм «зіставлення точки з лінією». Якщо v_i ближчий до $[\boldsymbol{H_r^n E_r^{n-1}}]$ ніж до $[\boldsymbol{H_l^n E_l^{n-1}}]$, v_i додається до $\boldsymbol{V_{farmr}^n}$. Інакше v_i додається до $\boldsymbol{V_{farml}^n}$. Основна вісь $\mathcal{E}\boldsymbol{V_{farm\,r}^n}$, $\mathcal{E}\boldsymbol{V_{farm\,l}^n}$ та L_{farm} використовуються для розрахунку$\boldsymbol{E_r^n}$ та $\boldsymbol{E_l^n}$.

Одна рука або з'єднані руки. $\boldsymbol{V_{hand}^n}$ містить тільки один компонент connex і відповідає сполученим рукам або тільки одній руці (іншу не видно). Використовуємо тимчасову узгодженість, щоб усунути ці два випадки.

Якщо $\boldsymbol{H_r^{n-1}}$ та $\boldsymbol{H_l^{n-1}}$ близькі до $\boldsymbol{V_{hand}^n}$, то руки поєднуються (рис.4.5 (b)) і $\boldsymbol{H_r^n} = \boldsymbol{H_l^n}$=Cog($\boldsymbol{V_{hand}^n}$) та ми обчислюємо $\boldsymbol{V_{farm}^n}$, як було запропоновано раніше. Сегментуємо $\boldsymbol{V_{farm}^n}$ в двох частинах $\boldsymbol{V_{farmr}^n}$ та $\boldsymbol{V_{farml}^n}$ ортогональною площиною до $[\boldsymbol{E_r^{n-1}E_l^{n-1}}]$ яка містить $\boldsymbol{H_l^n}$. Основну вісь $\mathcal{E}\boldsymbol{V_{farm\,r}^n}$, $\mathcal{E}\boldsymbol{V_{farm\,l}^n}$ та L_{farm} використовують для розрахунку $\boldsymbol{E_r^n}$ і $\boldsymbol{E_l^n}$.

В іншому випадку для обчислення сторони $\boldsymbol{H_x^n}$ та $\boldsymbol{H_x^n}$= $Cog(\boldsymbol{V_{hand}^n})$ використовується найближча рука $\boldsymbol{H_x^{n-1}}$ до $\boldsymbol{V_{hand}^n}$. Ми обчислюємо руку $\boldsymbol{V_{farm}^n}$, як було запропоновано раніше, і її головна вісь інерції використовується для обчислення $\boldsymbol{E_x^n}$.

Немає видимих рук. $\boldsymbol{V_{hand}^{n}}$ порожній, жодної руки не видно. Повертаємо позиції, обчислені на n - 1 кадрі до поточного кадра.

У всіх випадках $\boldsymbol{V_{act}^{n}}$ оновлюється, видаляючи елементи, що належать передпліччям або рукам.

Відстеження плечей. Оскільки руки знаходяться в сфері, зосередженій на нижній частині голови, з радіусом $L_{stat}/2$, то вокселі $\boldsymbol{V_{act}^{n}}$, які знаходяться в цій сфері, містять вокселі рук і шум вокселей. Нехай $\boldsymbol{V_{arms}^{n}}$ - множина цих вокселей.

Лікті знаходяться на одному кінці рук, таким чином, другий кінець рук відповідає плечам. Поточна позиція ліктя відома, далі визначаємо воксель.

Нехай $\boldsymbol{V_{armx}^{n}}$ (де *x* відповідає стороні) - найближча компонента коннектора $\boldsymbol{V_{arms}^{n}}$ до $\boldsymbol{E_{x}^{n}}$. Крім того, довжина руки L_{arm} є постійною, тоді поточний стан плеча $\boldsymbol{S_{x}^{n}}$ для сторони *x* визначається наступним чином:

$$\boldsymbol{S_{x}^{n} = E_{x}^{n} + \frac{Cog\,(V_{armx}^{n}) - E_{x}^{n}}{|Cog\,(V_{armx}^{n}) - E_{x}^{n}|}\, L_{arm}} \qquad (4.2)$$

Алгоритм відстеження плечей демонструє (рис.4.5 (c)). $\boldsymbol{V_{act}^{n}}$ оновлюється шляхом видалення елементів, що належать кожному плечу.

Відстеження ніг. Нехай $\boldsymbol{V_{act}^{n}}$ містить вокселі ніг. Для вилучення суглобів ніг був використаний процес «зіставлення точки з рядком», використовуваним для прив'язки скелета анімації на тривимірній сітці [39].

Елементи $\boldsymbol{V_{act}^{n}}$ розділені на чотири групи $\boldsymbol{V_{\mathbf{thigh}l}^{n}, V_{\mathbf{calf}\,l}^{n}, V_{\mathbf{thigh}r}^{n}}$, и $\boldsymbol{V_{\mathbf{calf}r}^{n}}$ в залежності від їх евклідової відстані до відрізків $\boldsymbol{\left[P_{l}^{n-1}, K_{l}^{n-1}\right]}$, $\boldsymbol{\left[K_{l}^{n-1}, F_{l}^{n-1}\right]}$$\boldsymbol{[P_{r}^{n-1}, K_{r}^{n-1}]}$ и $\boldsymbol{[K_{r}^{n-1}, F_{r}^{n-1}]}$ (рис. 4.5 (a)). Для лівої / правої сторони *x* обчислюємо еліпсоїд інерції $\boldsymbol{\mathcal{E}V_{calf\,x}^{n}}$ (нехай Ex_0 та Ex_1 - його екстремальні точки) і еліпсоїд інерції $\boldsymbol{\mathcal{E}V_{thigh\,x}^{n}}$

Коліно є точкою перетину між стегном і гомілками (рис.4.6 (b)), тому

положення стопи $\boldsymbol{F_x^n}$ визначається точкою нахилу $\mathcal{E}\boldsymbol{V_{calf\,x}^n}$, найвіддаленішої від еліпсоїда інерції $\boldsymbol{V_{\mathbf{thigh}x}^n}$, (наприклад Ex_1). Потім коліно вирівнюється по $[Ex_0Ex_1]$, Ex_0 сторона, на відстані L_{calf} від $\boldsymbol{F_x^n}$.

Положення тазостегнового суглоба $\boldsymbol{P_x^n}$ задається найдальшою точкою екстремуму $\mathcal{E}\boldsymbol{V_{thigh\,x}^n}$ від еліпсоїда інерції $\boldsymbol{V_{\mathbf{calf}\,x}^n}$, коригується, щоб бути на відстані L_{thigh} від $\boldsymbol{K_x^n}$.

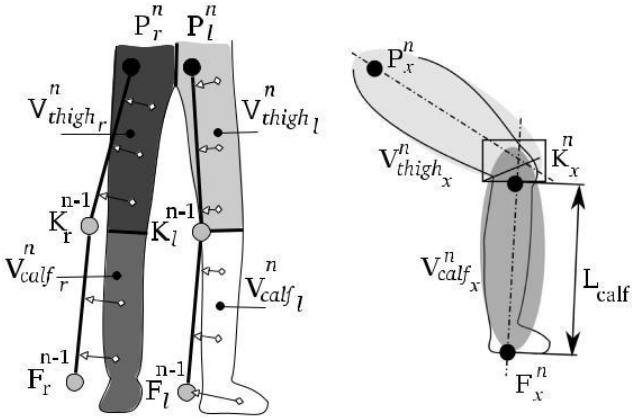


Рисунок 4.6 – етап «прив’язки» відстеження ніг і (б) оцінка суглобів ніг

4.5 Ініціалізація частин тіла

У даному пункті наводяться методи оцінки антропометричних мір і початкової пози тіла. Є можливість виконати класифікацію антропометричних мір за трьома методами, поданих в літературі щодо визначення початкової пози. Перший вид [40], антропометричні вимірювання та початкова поза вводяться вручну. Інших класів методів потрібна фіксована поза, така як T-pose [41], ці методи працюють в режимі реального часу. Останній клас методів повністю автоматичний [38] і не потребує конкретної позі, але не в режимі реального часу.

Запропонований для використання підхід призначений для роботи в режимі реального часу і повністю автоматизований для будь-якого руху, поки людина, що відзнята, стає, руки нижче рівня голови, а ноги не з’єднуються.

Після антропометричних оцінок метод послідовно обчислює кожен параметр частин тіла відповідно до кроків відстеження.

Антропометричні вимірювання. Для розробки оцінок коефіцієнтів використовуються спрощені антропометричні співвідношення, точність яких є достатньою для людино-машинних взаємодій. Нехай L_{stat} буде придбаною довжиною людського тіла, оціненою як максимальна відстань від передніх вокселов до площини підлоги. Таким чином, знаючи L_{stat}, вгадує для антропометричних показників наведені в цих співвідношеннях:

$R_{head} \approx L_{stat}/16$, $L_{torso} \approx 3\, L_{stat}/8$, $L_{calf} \approx L_{stat}/4$,

$L_{farm} \approx L_{stat}/6$, $L_{arm} \approx L_{stat}/6$, $L_{thigh} \approx L_{stat}/4$.

Як і в кроці відстеження, активний набір вокселей V_{act} ініціалізуються усіма вокселями $V_{all.}$

Ініціалізація голови. Цей крок спрямований на пошук T^0 та B^0. Вокселі обличчя $\boldsymbol{V^0_{face}}$ отриманого предмета визначаються самим верхнім компонентом зв'язку між $\boldsymbol{V^0_{skin}}$. Потім алгоритм відстеження голови застосовується для обчислення T^0 та B^0 без оцінки положення позиції обличчя. $\boldsymbol{V^0_{act}}$ оновлюється шляхом видалення елементів, що належать $\boldsymbol{V^0_{face}}$.

Ініціалізація торса. Алгоритм торса застосовується з використанням $\boldsymbol{V^0_{act}}$ в якості початкового значення для $\boldsymbol{V^0_{torso}}(0)$. $\boldsymbol{D^0_{torso}}(0)$. ініціалізується як вектор з N^0 в центр тяжіння $\mathcal{E}\,\boldsymbol{V^0_{act}}(0)$. Потім обчислюються положення P^0, $\boldsymbol{D^0_{t2d}}$ и $\boldsymbol{D^0_{l2r}}$. $\boldsymbol{V^0_{act}}$ оновлюється шляхом видалення елементів, що належать $\boldsymbol{V^0_{torso}}$.

Ініціалізація рук. Для ініціалізації позиції рук і передпліччя був використаний алгоритм відстеження рук. Оскільки немає попередніх позицій рук, можна тільки обчислити позиції передпліччя, коли є два різних передпліччя. Перевіривши ці критерії, можемо обчислити $\boldsymbol{H^0_r}$, $\boldsymbol{H^0_l}$, $\boldsymbol{E^0_r}$ та $\boldsymbol{E^0_l}$. $\boldsymbol{V^0_{act}}$ оновлюється шляхом видалення елементів, що належать до передпліччю. Позиції плечей $\boldsymbol{S^0_r}$ та $\boldsymbol{S^0_l}$ ініціалізуються безпосередньо з використанням алгоритму відстеження плечей, що був розглянутий вище.

Ініціалізація ніг. Алгоритм відстеження ніг потребує попередньо визначеної позиції ніг. Для цього імітуємо їх грубою оцінкою суглобів колін, ніг і стегон, потім обчислюємо більш точне положення сполучень ніг, з використанням алгоритму відстеження ніг. $\boldsymbol{V_{act}^0}$ містить воксели, які не використовувалися ні для яких інших частин тіла. Спочатку обчислимо набір компонентів коннекс з елементів $\boldsymbol{V_{act}^0}$, які мають їх висоту нижче $L_{stat} / 8$. Якщо є менше двох компонентів коннекс, припускаємо, що ноги з'єднані і не можуть бути виділені. В іншому випадку ми використовуємо дві основні компоненти зв'язку $\boldsymbol{V_{foot\,l}^0}$ та $\boldsymbol{V_{foot\,r}^0}$. Ліве і праве привласнення набору вокселей виконується з використанням вектора зліва-направо $\boldsymbol{D_{l2r}}$.

Для лівої / правої сторони x, нехай v_x - вектор з P^0 в центр тяжіння $\boldsymbol{V_{foot\,x}^0}$. Суглоби коліна і стопи визначаються з використанням наступних рівнянь:

$$\boldsymbol{K_z^{-1} = P^0 + v_x \frac{L_{thigh}}{|v_z|}}$$

$$\boldsymbol{F_z^{-1} = P^0 + v_x \frac{L_{thigh} - L_{calf}}{|v_z|}}$$

Оцінюємо попередні позиції стегна $\boldsymbol{P_l^{-1}}$ та $\boldsymbol{P_r^{-1}}$ як P_0. Нарешті, обчислюємо $\boldsymbol{F_r^0, K_r^0, F_l^0, K_l^0}$ використовуючи алгоритм відстеження ніг.

4.6 Результати застосування системи

Наведемо результати запропонованої системи. На рис.4.2(а) показана конфігурація системи. Інфраструктура придбання складається з чотирьох веб-камер Phillips (SPC900NC), підключених до одного ПК (CPU (центральний процесор): P4 3.2ghz, GPU (графічний процесор): NVIDIA Quadro 3450). Веб-камери створюють зображення з роздільною здатністю 320 × 240 при 30 кадрах в секунду.

Бібліотека IEEE 1394, розроблена в університеті Карнегі-Меллона

[UBNN], використовується для керування камерами, і бібліотека OpenCV [DHF+] використовується для калібрування камер. Програма написана середовищі розробки програмного забезпечення Visual Studio.

Метод застосовувався для різних осіб, що виконують швидкі і складні рухи. Використовуючи аналіз форми і кольору шкіри, алгоритм може обробляти складні пози, як показано на рис.4.7 (a). Ця поза ускладнена, тому що топологія тривимірної відновленої форми не є когерентною з топологією тривимірної фігури людини. Часова когерентність є ключем до успіху для пози, представленої на рис. 4.7 (b).

Це підкреслює випадок суміщених рук, який успішно розпізнається. Дуже складна поза показана на рис.4.7(c) і повністю відновлюється системою. Зображення, показані на рис.4.8, показує, що система працює для великого діапазону рухів.

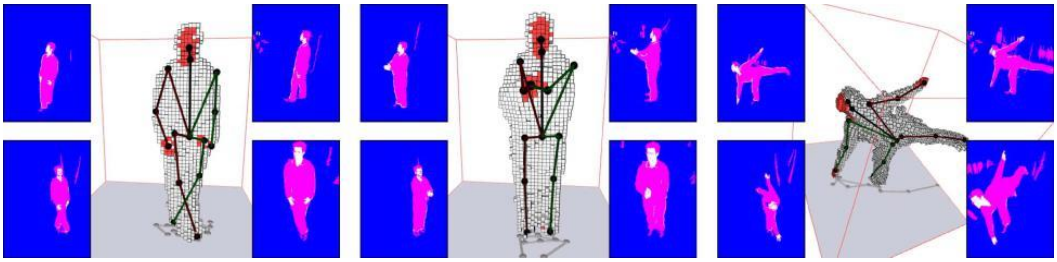


Рисунок 4.7 – (a) (b) та (c) підкреслюють результати для складних поз. Користувач відновленої пози представлений у вигляді анімації скелета, який має правосторонні частини в червоному кольорі і лівосторонні частини в зеленому кольорі.

Поточна експериментальна реалізація може відстежувати більше 30 поз в секунду на одному комп’ютері, що швидше, ніж частота кадрів прийому веб-камер. Оптимізована реалізація може бути використана для сучасного покоління домашніх комп’ютерів. Оскільки алгоритм заснований на 3D-реконструкції, він не залежить від кількості використовуваних камер, але

залежить від дозволу сітки вокселей.

Реконструйовано воксельного сітку, що складається з 64^3 вокселей в коробці розміром 6 $м^3$, яка дає приблизну роздільну здатність 2,7 × 2,7 × 2,7 см на воксель. Цієї роздільної здатності для людино-машинних інтерфейсів в сфері розваг.

У таблиці 4.1 показано порівняння швидкості з іншими поточними методами відстеження в реальному часі. Крім того, даний підхід забезпечує найкращу частоту кадрів тільки з одним комерційним ПК і оригінальною повністю автоматизованою і в реальному часі ініцілізацією.

Таблиця 4.1 – Порівняння показників представленого метода з існуючими методами

Посилання на методи (пункти)	Кількість суглобів	Ініціалізація в реальному часі	Відстеження частоти кадрів
4	15	Так	≈ 30кадрів / с
3.3	15	Ні	≈15 кадрів / с
3.1	19	Ні	≈10 кадрів / с
3.4	15	Ні	≈ 1 кадрів / с

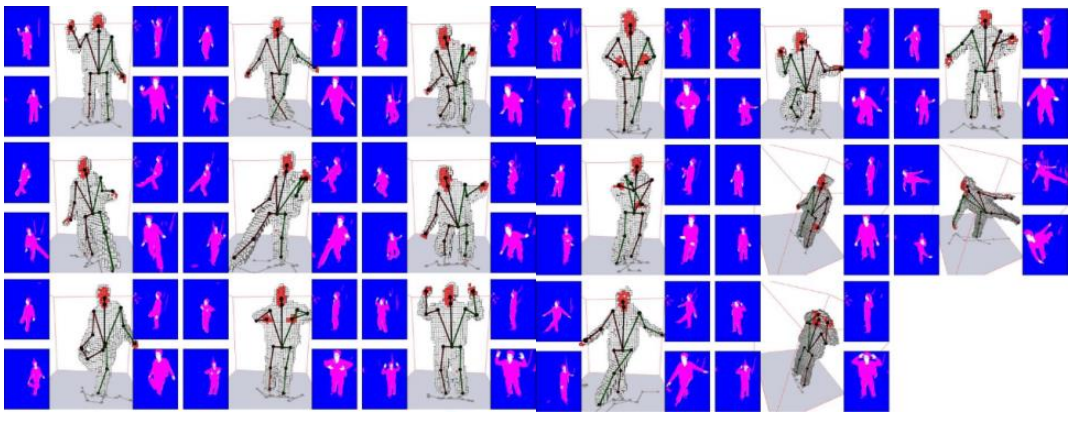


Рисунок 4.8 – Результати для широкого діапазону рухів

Система захоплення руху заснована на алгоритмі Shape-From-Silhouette. Цей алгоритм обчислює оцінку 3D-фігури об’єкта за своїми силуетами. Результат прямо залежить від якості сегментації силуету, який завжди є відкритою проблемою науки комп’ютерного зору.

Якщо в масці силуету є деякі шуми, такі як шум камери або тіні об’єктів, відновлення обсягу буде дуже зашумленим. Таким чином, результати захоплення руху будуть гірше. Але представлений метод також заснований на сегментації шкіри, який більш стійкий до шуму камери. Тоді суглоби рук і голови більш стійкі до шуму, ніж інші зчленування.

Сегментація, яка була обрана, заснована на стохастичному вивченні кольору з кольоровим набором зображень. Важливо, щоб процес міг навчатися, спираючись на великий набір даних, з різного роду зразків шкіри. Якщо вибірка шкіри упереджена, система буде давати гірші результати, особливо коли колір шкіри людини, що знімається не вивчається.

Висновки до розділу

У цьому розділі запропонована нова безмаркерна система захоплення руху людини, яка працює з декількома камерами (три або більше) і одним комп’ютером. Система заснована на аналізі тривимірної фігури, обмеженні морфології людини і сегментації шкіри 3D-форми. Вона повністю автоматизована і працює в режимі реального часу. Об’єднуючи різноманітну тривимірну інформацію, підхід є стійким до самоокклюзії. Він оцінює п’ятнадцять основних суглобів людського тіла зі швидкістю більше 30 кадрів в секунду. Порівняно з методами, наведеними у третьому розділі, представлений метод має простіщі алгоритми, вищу швидкість відстеження частоти кадрів та проводить ініціалізацію у реальному часі.

5 СТАРТАП-ПРОЕКТ

5.1 Опис ідеї проекту

Таблиця 5.1 – Опис ідеї стартап-проекту

Зміст ідеї	Напрямки застосування	Вигоди для користувача
Відкриття студії кіноманімації для потреб кіноіндустрії та навчання створення 3D моделей.	1. Освіта	Можливість студентам дізнатись більше про анімацію зйомки руху.
	2. Кіно та відеоігри	Можливість реалізації створення відео та ігор з тривимірним персонажем
	3. Медицина	Перевірка правильності рухів та положення тіла при виконанні фізичних вправ або за роботою.

Таблиця 5.2 – Визначення характеристик ідеї проекту

№ п/п	Техніко-економічні характеристики ідеї	(потенційні) товари/концепції конкурентів			W (слабка сторона)	N (нейтральна сторона)	S (сильна сторона)
		Запропонований метод	NANSEN SE	Mocap One			
1.	Пропозиція продажу або оренди професійного обладнання	Дає змогу	Дає змогу	Не дає змогу		+	
2.	Вартість послуги За день	250$	1799$	1000$			+
3.	Система	Безмаркерна	Маркерна	Маркерна			+

5.2 Технологічний аудит ідеї проекту.

У таблиці 5.3 показано оцінку технологічної здійсненності ідеї проекту та наведено технології, що можуть бути використані для реалізації проекту.

Таблиця 5.3. Технологічна здійсненність ідеї проекту

№ п/п	Ідея проекту	Технології її реалізації	Наявність технологій	Доступність технологій
1	Студія кіноанімації	Спеціалізоване обладнання для безмаркерного захоплення руху	Наявна	Доступна
2		Застосування апаратних систем	Необхідно розробити	Доступна
3		Розробка власних апаратно-програмних рішень	Наявна	При обмеженому бюджеті недоступна

Обрана технологія реалізації ідеї проекту: застосування спеціалізованого обладнання для проведення безмаркерного захоплення руху для анімації.

5.3. Аналіз ринкових можливостей запуску стартап-проекту

У таблиці 5.4 показано попередню характеристику потенційного ринку стартап-проекту.

Таблиця 5.4. Попередня характеристика потенційного ринку стартап-проекту

№ п/п	Показники стану ринку (найменування)	Характеристика
1	Кількість головних гравців, од	6
2	Загальний обсяг продаж, грн/ум.од	500 000
3	Динаміка ринку (якісна оцінка)	Зростає
4	Наявність обмежень для входу (вказати характер обмежень)	Зацікавлення потенційних клієнтів
5	Специфічні вимоги до стандартизації та сертифікації	Немає
6	Середня норма рентабельності в галузі (або по ринку), %	500000/210000 = 238%

У таблиці 5.5 показано характеристику потенційних клієнтів стартап-проекту.

Таблиця 5.5. Характеристика потенційних клієнтів стартап-проекту

№ п/п	Потреба, що формує ринок	Цільова аудиторія (цільові сегменти ринку)	Відмінності у поведінці різних потенційних цільових груп клієнтів	Вимоги споживачів до товару
1	Здешевлення процесу анімації 3D моделей	ЗМІ, медійні компанії	Рівень очікування якості захоплення руху	Відповідність результату найвищим стандартам якості
2	Пришвидшення процесу захоплення руху	ЗМІ, медійні компанії	Кожна з потенційних цільових груп має свої вимоги до плавності анімації	Забезпечення захоплення руху для анімації 3D моделей в залежності від рівня потреб споживача

У табл. 5.6 показані фактори загроз реалізації стартап-проекту.

Таблиця 5.6. Фактори загроз

№ п/п	Фактор	Зміст загрози	Можлива реакція компанії
1	Незацікавленість клієнтів	Внаслідок невдалого маркетингу клієнт може не зацікавитись послугами	Внесення додаткових сервісних послуг, демонстарція можливостей
2	Втрата конкуренції	Втрата рангу надійного поставника	Якісне та кількісне нарощування інтенсивності та грамотна цінова політика

У табл.5.7 показано фактори можливостей при реалізації стартап-проекту.

Таблиця 5.7. Фактори можливостей

№ п/п	Фактор	Зміст можливості	Можлива реакція компанії
1	Перехід до домінування на ринку медійних послуг	Зростання попиту	Якісне та кількісне нарощування потужностей
2	Імплементація технологій в існуючі системи захоплення руху	Зростання попиту внаслідок зростання клієнтів	Якісне та кількісне нарощування потужностей

У таблиці 5.8 визначено особливості конкурентного середовища та його вплив на впровадження проекту [42].

Таблиця 5.8. Ступеневий аналіз конкуренції на ринку

Особливості конкурентного середовища	В чому проявляється дана характеристика	Вплив на діяльність підприємства (можливі дії компанії, щоб бути конкурентоспроможною)
1.Чиста конкуренція	Використання схожих технологій	Стандартизація на високому рівні
2.Локальний	Відсутність єдиного національного постачальника послуг	Окремий підхід до кожної локальної ділянки
3.Міжгалузева	Відсутня	Відсутня
4.Товарно-видова	Застосування стандартизованих технологій	За необхідності, використання загальновживаних апаратних та програмних засобів
5.Цінова	Застосування спеціалізованих комплексів, які мають значну ціну	Можливість заощадити за допомогою застосування загальновживаних апаратних засобів
6.Марочна	Кожна діагностика має бути стандартизованою	Отримання переваги на ринку медійних послуг

У таблиці 5.9 показано аналіз конкуренції проекта в галузі за М. Портером

Таблиця 5.9. Аналіз конкуренції в галузі за М. Портером

Складові аналізу	Прямі конкуренти в галузі	Потенційні конкуренти	Постачальники	Клієнти	Товари-замінники
	Постачальники маркерних технологій	Необхідність пошуку постачальників	Залучення малопопулярних постачальників	Незалежність у прийнятті клієнтських рішень	Надання переваги більш авторитетним апаратним рішенням
Висновки:	Середня	Можливість виходу на ринок є	Постачальники диктують цінову політику на обладнання	Клієнти диктують вимоги до якості	Обмеження існують лише у разі відмови від діагностики

У табл. 5.10 показано фактори конкурентноспроможності та їх обґрунтування.

Таблиця 5.10. Обґрунтування факторів конкурентоспроможності

№ п/п	Фактор конкурентоспроможності	Обґрунтування (наведення чинників, що роблять фактор для порівняння конкурентних проектів значущим)
1	Раціональніший ціновий показник	Можливість більш раціонально використати ресурси на покращення якості захоплення руху
2	Надання сервісних послуг	Сервісна підтримка програмної частини

У табл. 5.11 наведено сильні та слабкі сторони проекту.

Таблиця 5.11. Порівняльний аналіз сильних та слабких сторін проекту

№ п/п	Фактор конкурентоспроможності	Бали 1-20	Рейтинг товарів-конкурентів у порівнянні						
			–3	–2	–1	0	+1	+2	+3
1	Раціональніший ціновий показник	17	+						
2	Надання сервісних послуг	12		+					
3	Періодична діагностика	4				+			
4	Необхідність залучення висококваліфікованих кадрів	7							+

У табл.5.12 наведено SWOT-аналіз стартап-проекту.

Таблиця 5.12. SWOT- аналіз стартап-проекту

Сильні сторони: раціональний ціновий показник, надання сервісних послуг	Слабкі сторони: періодична діагностика, можливості погрішностей при захопленні руху
Можливості: Перехід до ексклюзивного застосування нового методу, Імплементація методу в існуючі комплекси захопленя руху	Загрози: Незацікавленість клієнтів, втрата авторитету

Альтернативи ринкового впровадження стартап-проекту наведені у

табл.5.13.

Таблиця 5.13. Альтернативи ринкового впровадження стартап-проекту

№ п/п	Альтернатива (орієнтовний комплекс заходів) ринкової поведінки	Ймовірність отримання ресурсів	Строки реалізації
1	Укладення договорів з медійними компаніями та швидке захоплення ринку при використанні нового рішення	висока	незначні
2	Використання приладів загального вжитку для підвищення конкурентноспроможності	середня	незначні

Обрана альтернатива - укладення договорів з медійними компаніями та швидке захоплення ринку при використанні нового рішення

5.4. Розроблення ринкової стратегії проекту

Обгрунтування вибору цільових груп потенціійних споживачів наведено у табл. 5.14 [42].

Таблиця 5.14. Вибір цільових груп потенційних споживачів

№ п/п	Опис профілю цільової групи потенційних клієнтів	Готовність споживачів сприйняти продукт	Орієнтовний попит в межах цільової групи (сегменту)	Інтенсивність конкуренції в сегменті	Простота входу у сегмент
1	Медійні та кінокомпанії	Середня	Високий	Середня	Висока
2	Аматорські кіностудіїї та рекламні агенства	Висока	Високий	Середня	Низька

Визначення базової стратегії розвитку наведено у табл. 5.15.

Таблиця 5.15. Визначення базової стратегії розвитку

№ п/п	Обрана альтернатива розвитку проекту	Стратегія охоплення ринку	Ключові конкурентоспроможні позиції відповідно до обраної альтернативи	Базова стратегія розвитку*
1	Використання альтернативних технологій та пристроїв	Встановлення нового стандарту якості	Зацікавлення та залучення гігантів у галузі телебачення та кіно	Стратегія диференці ації
2	Дешевизна проекту	Раціональніші витрати на обладнання, та послуги	Застосування загальновживаних апаратних рішень замість спеціалізованих комплексів	Стратегія лідерства по витратах

Визначення базової стратегії конкурентної поведінки наведено у табл.5.16.

Таблиця 5.16. Визначення базової стратегії конкурентної поведінки

№ п/п	Чи є проект «першопрохідцем» на ринку?	Чи буде компанія шукати нових споживачів, або забирати існуючих у конкурентів?	Чи буде компанія копіювати основні характеристики товару конкурента, і які?	Стратегія конкурент ної поведінки *
1	Так	Забирати існуючих та шукати нових	Не буде	Стратегія виклику лідера

Визначення стратегії позиціонування наведено у табл. 5.17.

Таблиця 5.171. Визначення стратегії позиціонування

№ п/п	Вимоги до товару цільової аудиторії	Базова стратегія розвитку	Ключові конкурентоспромо жні позиції власного стартап-проекту	Вибір асоціацій, які мають сформувати комплексну позицію власного проекту (три ключових)
1	Висока якість послуг	Стратегія диференціації	Новизна, гарант якості, точність дослідження	Якість, надійність, точність
2	Мінімальні витрати	Стратегія лідерства по витратах	Універсальність запропонованого рішення	Дешевизна, універсальність

5.5. Розроблення маркетингової програми стартап-проекту

Ключові переваги концепції потенційного товару наведено у табл. 5.18.

Таблиця 5.18. Визначення ключових переваг концепції потенційного товару

№ п/п	Потреба	Вигода, яку пропонує товар	Ключові переваги перед конкурентами (існуючі або такі, що потрібно створити
1	Якість	Висока якість, надійність	Надійність
2	Дешевизна	Раціональне використання коштів, дешевше обладнання	Дешевизна

Визначено три рівні моделі товару. Сутність та складові рівнів товару наведено у табл. 5.19.

Таблиця 5.19. Опис трьох рівнів моделі товару

Рівні товару	Сутність та складові		
I. Товар за задумом	Якісні послуги, стандартизована якість послуг та обладнання		
II. Товар у реальному виконанні	Властивості/характеристики	М/Нм	Вр/Тх /Тл/Е/Ор
	1)Вартість обслуговування, 2)Кількість комплектів обладнання 3)Строк безвідмовної праці 4)Технологічна собівартість товару	1) М 2) М 3) М 4) М	1)Е 2) Пр 3)Нд 4)Тх
	Якість: міжнародні стандарти якості, постійна підтримка обладнання		
	Доставка, встановлення та налаштування		
	Марка: Кіновиробництво		
III. Товар із підкріпленням	До продажу – обладнання, встановлення		
	Після продажу – сервісна підтримка		

За рахунок чого потенційний товар буде захищено від копіювання: специфічна методика захоплення та обробка даних.

Визначення меж встановлення ціни на послугу наведено у табл. 5.20.

Таблиця 5.20. Визначення меж встановлення ціни

№ п/п	Рівень цін на товари-замінники	Рівень цін на товари-аналоги	Рівень доходів цільової групи споживачів	Верхня та нижня межі встановлення ціни на товар/послугу
1	2500 у.о./од.	1800 у. о./од	Високий	Н.500 у.о. – В.1000 у.о. (Товар) Н.200 у.о. – В.500 у.о. (Послуга)

Формування системи збуту послуги наведено у табл. 5.21.

Таблиця 5.21. Формування системи збуту

№ п/п	Специфіка закупівельної поведінки цільових клієнтів	Функції збуту, які має виконувати постачальник товару	Глибина каналу збуту	Оптимальна система збуту
1	Орієнтована на отримання максимальної якості та точності захоплення руху	Поставки якісного, точного та надійного товару	Значна	Договірна система збуту

Концепції маркетингових комунікацій наведено у табл. 5.22.

Таблиця 5.22. Концепція маркетингових комунікацій

№ п/п	Специфіка поведінки цільових клієнтів	Канали комунікацій, якими користуються цільові клієнти	Ключові позиції, обрані для позиціонування	Завдання рекламного повідомлення	Концепція рекламного звернення
1	Зацікавленість в якісному та точному продукті з раціональним використанням ресурсів	Медіа ресурси	Гарантованість якості та стандартизація, політика сервісності	Зацікавити у покращеннях пов’язаних із зростаючою популярністю послуг	Представлення центру створення реалістино рухомих тривимірних моделей
2	Зацікавленість у великій кількості продукту із дотриманням умов якісності	Медіа ресурси	Глибина каналу постачальників, гарант якості	Зацікавити у позитивних сторонах первісності та в глибині каналу постачання	Представлення центру створення реалістино рухомих 3D та 2D моделей

Висновки до розділу

1. Комерціалізацію стартап-проекту щодо розвитку та впровадження запропонованого апаратно-програмного рішення для створення анімації тривимірних моделей шляхом безмаркерного захоплення руху, можна вважати доцільною. На дану пропозицію на ринку медійних послуг присутній попит, наразі він задовольняється товарами замінниками та більш дорогими рішеннями, саме тому важливо зайняти нішу конкурента у якості поставника вигідного продукту, порівнюючи з конкурентами. Рентабельність на ринку послуг насамперед обумовлена заміною повної апаратної залежності на універсальність, що обумовлена використанням не спеціалізованих комплексів, а загальновживаного програмного та апаратного забезпечення.

2. Впровадження є перспективним, адже основними групами клієнтів є масштабні телевізійні та кіно компанії, і після набуття достатньої авторитетності можливе охоплення у масштабах міжнаціональних ринків. Конкурентноспроможність проекту обумовлена меншою ціною на повний продукт та високою якістю створення motion capture в умовах, коли конкуренти за цим параметром у даних умовах програють. Це вигідно вирізняє запропоноване рішення, власне, і є основним критерієм входження на ринок.

3. Альтернативою впровадження було обрано – пошук альтернативних технологій та пристроїв для побудови систем створення схожих відео. Імплементація проекту доцільна, оскільки рентабельність та зацікавленість потенційних груп клієнтів створює досить сприятливі умови для розвитку проекту.

ВИСНОВКИ

У магістерській дисертації запропоновано метод безмаркерного захоплення руху, з використанням доступного обладнання, що забезпечує швидкодію обробки даних та якісні результати. Метод зрозумілий для звичайних користувачів і може мати практичну цінність для навчальних потреб, зокрема, для реалізації кваліфікаційних та атестаційних робіт і проектів студентів кафедри.

В рамках магістерської дисертації було проведено дослідження безмаркерних методів захополення руху. На основі проведених досліджень отримано наступні результати:

1. Проаналізовано теоретичні засади комп'ютерного розпізнання образів. Проведено огляд сучасних технологій безмаркерного захоплення руху. Вивчено та описано алгоритми розпізнавання образів, а саме позиціонування з використанням зображення просторової глибини з побудовою 3D топологічної сітки на поверхні тіла, з використанням дерева прийняття рішень та функції зображення з визначенням форми з силуету.

2. Розглянуто апаратно-програмні засоби для захоплення руху, такі як Microsoft Kinect та системи iPi Soft. Наведено приклади реалізації студії з використанням розробок Kinect та iPi Soft.

3. Вивчено особливості застосування трьох методів безмаркерної технології захоплення руху. Наведено опис методів, детально описані алгоритми засновані на оцінці ймовірності, за якими відбуваються відстеження моделі.

Встановлено, що у першому методі було користано набір тренажерів зі спіральним проходженням з 343 прикладами, що дозволяє визначити нові пози при середній швидкості ~ 0,104 секунди на кадр на Pentium TM 4 з процесором 2,8 ГГц. Вихідний потік попереднього зображення порівнюється з оригінальною позою, яка була використана для створення штучних силуетів. Метод забезпечує хороші результати, якщо кожна сторона тіла однозначно відображається для

оцінки тривимірної форми. З випадками самооклюзії або великими контактами між кінцівками та тілом метод часто не працює належним чином.

Другий метод базується на 4 фотоапаратах, що знімають зображення на швидкості 30 кадрів в секунду з роздільною здатністю 320×240. Система, включає в себе зйомку зображень, об'ємну реконструкцію та байєсову систему відстеження, працює на частоті 10 кадрів в секунду на одному комп'ютері з частотою 2 ГГц. Метод потребує використання форми та кольору. Такий метод вимагає використання контрастного одягу між кожною частиною тіла для коректного відстеження.

Третій метод, в якому розглянуто технологія до відновлення рухів тіла з різних видів за допомогою 3D скелетної моделі. Складність скелетнізації полягає в обчисленні клітин Вороного. Це займає близько 60 мс для 2000 поверхневих точок на Opteron 2 ГГц. Розподіл його обчислень дозволяє цей процес запускати зі швидкістю 30 кадрів на секунду. Потужність в режимі реального часу - менш ніж 30 мс. Методу потрібне ручне втручання для антропометричних вимірювань та оцінки початкової позиції.

Обгрунтовано, що перелічені методи не є достатньо зручними для користувача.

4. Запропоновано систему захоплення руху, що заснована на алгоритмі Shape-From-Silhouette. Алгоритм обчислює оцінку 3D-фігури об'єкта за силуетами. Для відстеження 3D-фігури об'єкта використовуються камерами (три або більше) і один комп'ютер. Система заснована на аналізі тривимірної фігури, обмеженні морфології людини і сегментації шкіри 3D-форми. Система повністю автоматизована і працює в режимі реального часу. Об'єднуючи різноманітну тривимірну інформацію, підхід є стійким до самоокклюзії. Він оцінює п'ятнадцять основних суглобів людського тіла зі швидкістю більше 30 кадрів в секунду. Також представлений метод заснований на сегментації шкіри, який більш стійкий до шуму камери.

5. Розроблено стартап-проект, який базується на просуванні на ринок студії кіноанімації, з використанням технології безмаркерного захоплення руху, що основана на застосуванні запропонованого методу. Проведено дослідження доцільності та рентабельності даного бізнес-проекту та визначено, що комерціалізація проекту є доцільною.

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАНЯ

1. Стаття «Технология Motion Capture»,
   URL: http://infoglaz.ru/41123-tehnologiya-motion.html (дата звернення 11.11.2017 р.)
2. Стаття «Markerless Motion»,
   URL: https://en.wikipedia.org/wiki/Motion_capture#Markerless (дата звернення 11.11.2017 р.)
3. Стаття «Технология распознования образов»,
   URL: http://www.rusnauka.com/2_KAND_2015/Informatica/3_185568.doc.htm (дата звернення 01.12.2017 р.)
4. Стаття «Дерево принятия решений и Случайный лес»,
   URL: https://proglib.io/p/ml-regression/ (дата звернення 05.12.2017 р.)
5. G. Haro, "Shape from silhouette consensus", Pattern Recognition, vol. 45, no. 9, pp. 3231-3244, 2012
6. Стаття «Visual hull»,
   URL:https://en.wikipedia.org/wiki/Visual_hull (дата звернення 14.12.2017 р.)
7. Стаття «Face Recognition with OpenCV»,
   URL: https://docs.opencv.org/2.4/modules/contrib/doc/facerec/facerec_tutorial.html (дата звернення 14.12.2017 р.)
8. Стаття «Basics of streaming protocols»,
   URL: http://www.garymcgath.com/streamingprotocols.html (дата звернення 15.12.2017 р.)
9. Стаття «Выделение и распознавание лиц» ,
   URL: http://wiki.technicalvision.ru/index.php/Выделение_и_распознавание_лиц (дата звернення 15.01.2018 р.)
10. Стаття «Markerless tracking»,
    URL: https://xinreality.com/wiki/Markerless_tracking (дата звернення 15.01.2018 р.)
11. Стаття «Microsoft Kinect»,
    URL: https://www.open-electronics.org/3d-scanning-with-microsoft-kinect/ (дата звернення 15.01.2018 р.)

12. Стаття «How does the Kinect work?» ,
URL: http://www.cs.bham.ac.uk/~vvk201/Teach/Graphics/kinect.pdf (дата звернення 17.01.2018 р.)
13. Стаття «Оснащення iPi Soft» ,
URL: http://ipisoft.com/ (дата звернення 17.01.2018 р.)
14. Стаття «Possible Configurations iPi Soft»,
URL: http://wiki.ipisoft.com (дата звернення 23.01.2018 р.)
15. Стаття «What are Memoji?» ,
URL: https://www.pocket-lint.com/phones/news/apple/144743-what-are-memoji-how-to-create-an-animoji-that-looks-like-you (дата звернення 06.02.2018 р.)
16. J.B. Tenenbaum, V. de Silva, and J.C. Langford. A global geometric framework for nonlinear dimensionality reduction. Science, 2904:2319–2323, 2000.
17. K. Weinberger and L. Saul. Unsupervised learning of image manifolds by semidefinite programming. In Int. Conf. on Computer Vision & Pattern Recognition, volume II, pages 988–995, 2004.
18. Y. Bengio, J.-F. Paiement, and P. Vincenta. Out of sample extensions for LLE, isomap, MDS, eigenmaps and spectral clustering. In Advances in Neural Information Processing Systems, volume 16, 2004.
19. B. Scholkopf,¨ A. Smola, and K. Muller¨. Kernel PCA and denoising in feature spaces. In Advances in Neural Information Processing Systems, pages 536–542, 1999.
20. L. Saul and S. Roweis. Think globally, fit locally: unsupervised learning of low dimensional manifolds. Journal of Machine Learning Research, 4:119–155, 2003.
21. B. Scholkopf,¨ S. Mika, A. Smola, G. Ratsch,¨ and K. Muller¨. Kernel PCA pattern re-construction via approximate pre-images. In International Conference on Artificial Neural Networks, pages 147–152, 1998.
22. K. Grauman and T. Darrell. The pyramid match kernel: Discriminative classification with sets of image features. In Int. Conf. on Comp. Vision, pages 1458–1465, 2005.
23. Стаття Метод Монте-Карло и его точность ,
URL: https://habr.com/post/274975/ (дата звернення 15.11.2017 р.)

24. Стаття Байесовские и марковские сети ,
URL: http://www.machinelearning.ru/wiki/images/5/5b/Lecture1_GM.pdf (дата звернення 15.11.2017 р.)
25. M. A. T. Figueiredo and A. K. Jain. Unsupervised learning of _nite mixture models. IEEE Trans. on PAMI, 24(3):381.396, 2002.
26. D. Ron, Y. Singer, and N. Tishby. The power of amnesia: Learning probabilistic automata with variable memory length. Machine Learning, 25(2.3):117.149, 1996.
27.A. Laurentini. The Visual Hull Concept for Silhouette-Based Image Understanding. IEEE Transactions on PAMI, 16(2):150±162, February 1994.
28.J. Serra. Image Analysis and Mathematical Morphology, Vol-ume I. Academic Press, 1982.
29.D. Attali and A. Montanvert. Modeling noise for a better simpli®cation of skeletons. In Proceedings of ICIP, Lau-sanne (Switzerland), 1996.
30. Brice Michoud, Erwan Guillou, and Sa¨ıda Bouakaz. Real-Time and Markerless 3D Human Motion Capture Using Multiple Views A. Elgammal et al. (Eds.): Human Motion 2007, LNCS 4814, pp. 88–103, 2007.
31. Zhang, Z.: Flexible camera calibration by viewing a plane from unknown orientations. In: ICCV, pp. 666–673 (1999)
32. Joshi, N.: Color calibration for arrays of inexpensive image sensors. Technical report, Stanford University (2004)
33. Toyama, K., Krumm, J., Brumitt, B., Meyers, B.: Wallflower: Principles and practice of background maintenance. In: ICCV (1), pp.255–261 (1999)
34. Hasenfratz, J.M., Lapierre, M., Sillion, F.: A real-time system for full body interaction with virtual worlds. In: Eurographics Symposium on Virtual Environments, pp. 147–156 (2004)
35. Vezhnevets, V., Sazonov, V., Andreeva, A.: A survey on pixel-based skin color detection techniques. In: Proceedings of Graphicon-2003 (2003)
36. Segal, M., Korobkin, C., van Widenfelt, R., Foran, J., Haeberli, P.: Fast shadows and lighting effects using texture mapping. In: Proceedings of SIGGRAPH (1992)
37. Culbertson, W.B., Malzbender, T., Slabaugh, G.G.: Generalized voxel coloring. In:Workshop on Vision Algorithms, pp. 100–115 (1999)

38. Mikic, I., Trivedi, M., Hunter, E., Cosman, P.: Human body model acquisition and tracking using voxel data. Int. J. Comput. Vision 53(3), 199–223 (2003)
39. Sun, W., Hilton, A., Smith, R., Illingworth, J.: Layered animation of captured data. The Visual Computer 17(8), 457–474 (2001)
40. M´enier, C., Boyer, E., Raffin, B.: 3d skeleton-based body pose recovery. In: Proceedings of the 3rd International Symposium on 3D Data Processing, Visualization and Transmission, Chapel Hill (USA) (June 2006)
41. Fua, P., Gruen, A., D'Apuzzo, N., Plankers, R.: Markerless Full Body Shape and Motion Capture from Video Sequences. In: Symposium on Close Range Imaging, International Society for Photogrammetry and Remote Sensing, Corfu, Greece (2002)
42. Методичні ре-комендації до виконання розділу магістерських дисертацій для студентів інженерних спеціальностей / За заг. ред. О.А. Гавриша. – Київ : НТУУ «КПІ», 2016. – 28 с.
43. Виноградча. Е.В. Можливості та перспективи використання технології доповненої реальності у сучасній освіті. К.: НТУУ «КПІ ім. Ігоря Сікорського», 2018. – С. 10.
44. Виноградча. Е.В. Дослідження можливостей безмаркерного захоплення руху з багатовидовим структурованим світлом. К.: НТУУ «КПІ ім. Ігоря Сікорського», 2018. – С. 5-10.
45. Виноградча. Е.В. Дослідження технології доповненої реальності в освіті та перспективи її застосування при вивчені електроніки. К.: НТУУ «КПІ ім. Ігоря Сікорського», 2018. – С. 41-45.

ДОДАТОК А

Реферат англійською мовою на тему магістерської дисертації

ABSTRACT

Motion capture (sometimes referred as mo-cap or mocap, for short) is the process of recording the movement of objects or people. It is used in military, entertainment, sports, medical applications, and for validation of computer vision and robotics. In filmmaking and video game development, it refers to recording actions of human actors, and using that information to animate digital character models in 2D or 3D computer animation. When it includes face and fingers or captures subtle expressions, it is often referred to as performance capture. In many fields, motion capture is sometimes called motion tracking, but in filmmaking and games, motion tracking usually refers more to match moving.

In motion capture sessions, movements of one or more actors are sampled many times per second. Whereas early techniques used images from multiple cameras to calculate 3D positions, often the purpose of motion capture is to record only the movements of the actor, not his or her visual appearance. This animation data is mapped to a 3D model so that the model performs the same actions as the actor.

Most modern systems can extract the silhouette of the performer from the background. Afterwards all joint angles are calculated by fitting in a mathematic model into the silhouette. For movements you can’t see a change of the silhouette, there are hybrid Systems available who can do both (marker and silhouette), but with less marker.

Emerging techniques and research in computer vision are leading to the rapid development of the markerless approach to motion capture. Markerless systems such as those developed at Stanford University, the University of Maryland, MIT, and the Max Planck Institute, do not require subjects to wear special equipment for tracking. Special computer algorithms are designed to allow the system to analyze multiple streams of optical input and identify human forms, breaking them down into constituent parts for tracking. ESC entertainment, a subsidiary of Warner Brothers Pictures created specially to enable virtual cinematography, including photorealistic digital look-alikes for filming The Matrix Reloaded and The Matrix Revolutions movies, used a technique called Universal Capture that utilized 7 camera setup and

the tracking the optical flow of all pixels over all the 2-D planes of the cameras for motion, gesture and facial expression capture leading to photorealistic results.

Markerless technologies use the features of the face such as nostrils, the corners of the lips and eyes, and wrinkles and then track them. This technology is discussed and demonstrated at CMU, IBM, University of Manchester (where much of this started with Tim Cootes, Gareth Edwards and Chris Taylor) and other locations, using active appearance models, principal component analysis, eigen tracking, deformable surface models and other techniques to track the desired facial features from frame to frame. This technology is much less cumbersome, and allows greater expression for the actor.

These vision based approaches also have the ability to track pupil movement, eyelids, teeth occlusion by the lips and tongue, which are obvious problems in most computer animated features. Typical limitations of vision based approaches are resolution and frame rate, both of which are decreasing as issues as high speed, high resolution CMOS cameras become available from multiple sources.

The technology for markerless face tracking is related to that in a Facial recognition system, since a facial recognition system can potentially be applied sequentially to each frame of video, resulting in face tracking. For example, the Neven Vision system (formerly Eyematics, now acquired by Google) allowed real-time 2D face tracking with no person-specific training; their system was also amongst the best-performing facial recognition systems in the U.S. Government's 2002 Facial Recognition Vendor Test (FRVT). On the other hand some recognition systems do not explicitly track expressions or even fail on non-neutral expressions, and so are not suitable for tracking. Conversely, systems such as deformable surface models pool temporal information to disambiguate and obtain more robust results, and thus could not be applied from a single photograph.

Markerless face tracking has progressed to commercial systems such as Image Metrics, which has been applied in movies such as The Matrix sequels and The Curious Case of Benjamin Button. The latter used the Mova system to capture a deformable facial model, which was then animated with a combination of manual and vision tracking. Avatar was another prominent performance capture movie however it

used painted markers rather than being markerless. Dynamixyz is another commercial system currently in use.

Markerless systems can be classified according to several distinguishing criteria:

- 2D versus 3D tracking
- whether person-specific training or other human assistance is required
- real-time performance (which is only possible if no training or supervision is required)
- whether they need an additional source of information such as projected patterns or invisible paint such as used in the Mova system.

To date, no system is ideal with respect to all these criteria. For example the Neven Vision system was fully automatic and required no hidden patterns or per-person training, but was 2D. The Face/Off system is 3D, automatic, and real-time but requires projected patterns.

Marker-free motion capture has long been studied in computer vision as classic and fundamental problems. While commercial real-time products using markers are already available, sound online marker-free systems remain an open issue because many real-time algorithms still lack robustness, or require expensive devices and time-consuming algorithms. While most popular techniques run on PC cluster, our system requires a small set of low-cost cameras (three or more) and a single computer. Our system works in real-time (30 fps), without markers (active or passive) or special devices.

We present an extension of Shape-From-Silhouette (SFS) algorithms. It recon-structs in real-time 3D shape and 3D skin-colored parts of a person from cali-brated cameras.

Usually, only SFS methods compute in real-time 3D shape estimation of an object, from its silhouette images. Silhouette images are binary masks corre-sponding to captured images where 0 corresponds to background, and 1 stands for the (interesting) feature of the object. The formalism of SFS was introduced by A. Laurentini. By definition, an object lies inside the volume generated by back-projecting its silhouette through the camera center (called silhouette’s cone). With multiple views of the same object at the same time, the intersec-tion of all the

silhouette's cones build a volume called "Visual Hull", which is guaranteed to contain the real object. There are mainly two ways to compute an object's Visual Hull.

Surface-Based Approaches. They compute the intersection of silhouette's cone surfaces. First silhouettes are converted into polygons. Each edge is back-projected to form a 3D polygon. Then each 3D polygon is projected onto each other's images, and is intersected with each silhouette in 2D. The resulting polygons are assembled to form an estimation of the polyhedral shape. Resulting Surface-based shape from silhouette is underlined. These approaches are not well suited to our application because of the complexity of the underlying geometric calculations. Furthermore incomplete or corrupted surface models could be created, directly depending upon polyhedron sharpness and silhouette noise.

Volumetric-Based Approaches. They usually estimate shape by processing a set of voxels. The object's acquisition area is split up into a 3D grid of voxels (volume elements). Each voxel remains part of the estimated shape if it projection in all images lies in all silhouettes. This volumetric approach is adapted for real-time pose estimation, due to its fast computation and robustness to noisy silhouettes.

We propose a new framework which computes a 3D volumetric shape and skin parts estimation on a single computer. The system consists of two tasks:

- Input data: Camera calibration data, silhouette and skin parts segmentation;
- 3D Shape and skin parts estimation: shape voxels are computed by a GPU SFS implementation and skin parts are determined using voxel visibility. Each task is described in their respective section.

Our current experimental implementation can track more than 30 poses per second on a single computer, which is faster than the webcams acquisition frame rate. An optimized implementation can be usable for current generation of home entertainment computers. As our algorithm is based on 3D reconstruction, it is independent of the number of cameras used, but it depends on the voxel grid resolution. We reconstruct a voxel grid composed by 643 voxels in a 6 m3 box which gives an approximate resolution of 2.7 × 2.7 × 2.7 cm per voxel. This resolution is enough for human-machine interfaces in the field of entertainment.

Our motion capture system is based on a Shape-From-Silhouette algorithm. This algorithm computes an object 3D shape estimation from its silhouettes. The result directly depends on the silhouette segmentation quality, which is always an opened problem of the computer vision science. If the silhouette mask contains some noises like camera noise or object shadows, the volume reconstruction will be very noised. Thus the results of the motion capture will be worse. But our method is also based on a skin segmentation which is a more robust faced to camera noise. Then the hand and head articulations are more noise-resistant, than others articulations.

The segmentation we have selected is based on a skin-colored stochastic learning from colored image set. It is important to make the leaning process on a big data set, with different kind of skin sample. If the skin sampling is biased then the system will provide worse results, especially when the skin color of the person filmed is not learned.